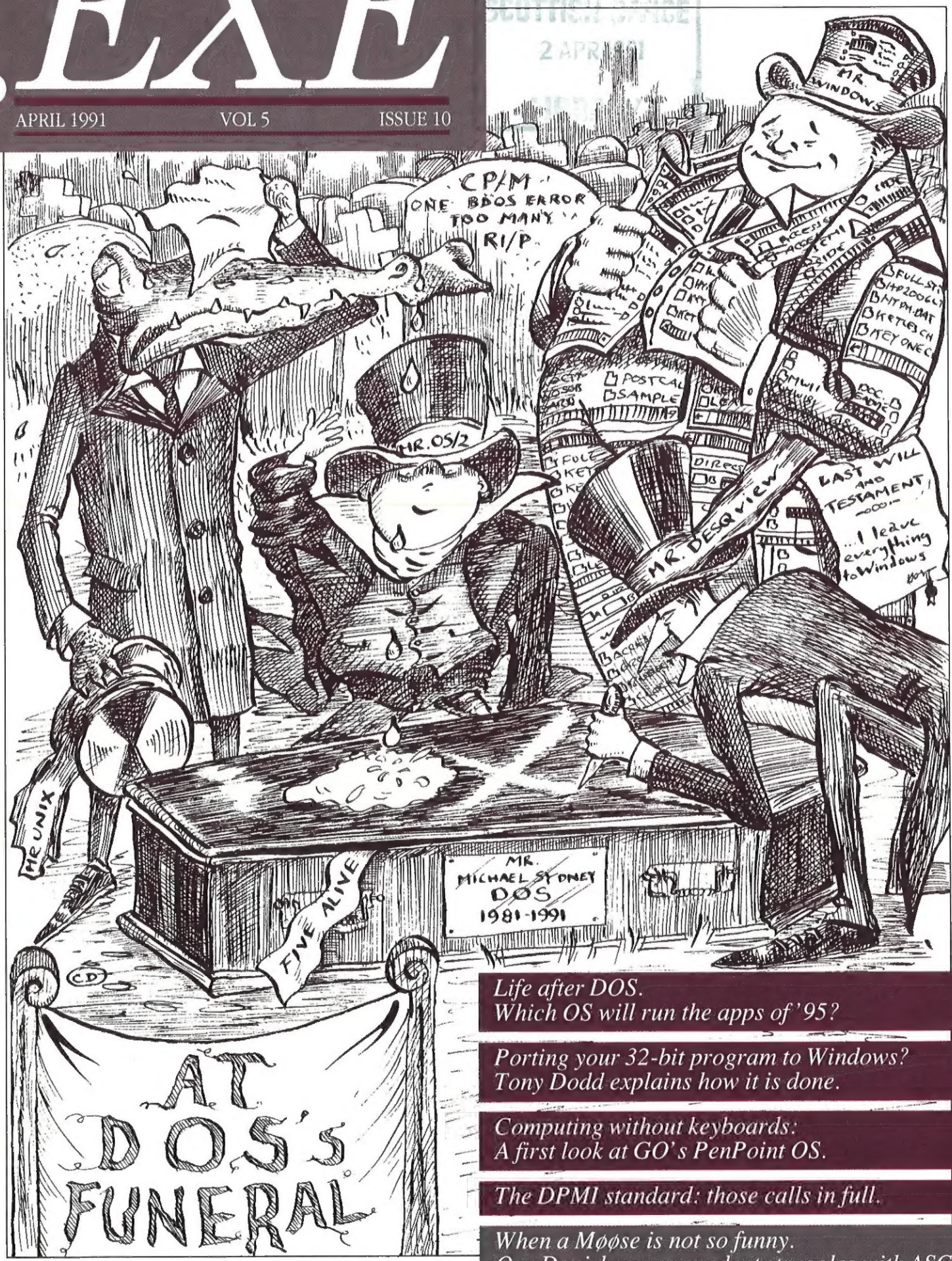
.EXE

The Software Developers' Magazine

APRIL 1991 VOL 5 ISSUE 10

WIN A TRIP TO CALIFORNIA
SEE PAGE SIX



Life after DOS.
Which OS will run the apps of '95?

Porting your 32-bit program to Windows?
Tony Dodd explains how it is done.

Computing without keyboards:
A first look at GO's PenPoint OS.

The DPMI standard: those calls in full.

When a Møøse is not so funny.
Our Danish correspondent struggles with ASCII.

THE NETWORKING LEADERS

The best run networks begin at QA

Configuring, operating, tuning and supporting end-user needs in networks all require indepth knowledge and specialised skills.

QA networking courses have been created precisely with those objectives in mind.

Whichever QA course you choose, it's based on our own extensive practical experience with all the PC industry's major networks. Our instructors are not only highly skilled in network system operation, but also outstanding in communicating these skills to others.

SUPPORTING PC NETWORKS
Network concepts and components. Network adapters and cable systems. Network operating systems. Installing and configuring applications. Performance tuning and fault finding. Security and data safety. Wide Area Networks. IBM SNA connectivity. *4 Days.* £760.

SUPPORTING NOVELL NETWORKS
NetWare architecture. Workstation shell. Server installation and configuration. Security and safety. Applications support. Printers. Performance and fault finding. External communications. Internetworking. Network protocols. *4 Days.* £760.

SUPPORTING IBM OS/2 AND DOS LANS
IBM LAN architectures. PC LAN program and DOS LAN requester. OS/2 E.E requester. IBM LAN Server installation and configuration. Applications support. Security and safety. Printers and devices. Performance and fault finding. External communications. *4 Days.* £760.

SUPPORTING OS/2 LAN MANAGER
OS/2 and LAN Manager architecture. DOS and OS/2 LAN Manager installation and configuration. Applications support. SQL. Security and safety. Printers and devices. Performance and fault finding. External communications. *4 Days.* £760.

SUPPORTING NETWARE 386
NetWare 386 architecture. New file and disk systems. Security enhancements. New user types. Printing. Protocol support. Quick installation. Performance tuning. Future strategies. *1 Day.* £220.

For more information, please complete and return the coupon today or call Samantha Trinder on 0285 655888.

Mail to QA Training Ltd, Cecily Hill Castle, Cirencester, Gloucestershire GL7 2EF. UK.

Tick here for full details:

☐ OS/2 ☐ Unix ☐ Windows
☐ C ☐ Object Oriented ☐ Networks
☐ Consultancy ☐ Products

NAME _____ TITLE _____

COMPANY _____

ADDRESS _____

TEL. NO. _____

EXE 4/91

QA TRAINING

Comprehensive Network Training in a Castle

CIRCLE NO. 645

Vol 5 Issue 10

.EXE

April 1991

Editor: Willie Watts
News Editor: Dan O'Brien
Editorial Assistant: Wendie Leek
Design & Layout: Katerina Adams, Mark English
Display Advertising Manager: Sandra Inniss-Palmer
Senior Ad Executive: Ed Butcher
Reader Services & Promotions: Helena Adams
Publisher: Jon Howell
Administration & Subscriptions: Rena Gibbs, Sandy McDonnell
Front Cover Illustration: Chris Duggan

General
.EXE Magazine is independent and not affiliated to any vendor of hardware, software or services. It is published by Process Communications Ltd, 10 Barley Mow Passage, Chiswick, London W4 4PH.
Tel: (Advertising/Editorial/Production) 081 994 6477
(Subscriptions) 044282 4501
Facsimile: 081 994 1533 Telex: 8811418 SPACES G
ISSN: 0268-6872

Subscriptions
.EXE Magazine is a monthly journal for software developers. It is available only by subscription, at a cost of £35 per annum (11 issues) in the UK (see insert card for academic and overseas rate). 'A subscription implies that this journal will be sent to the subscriber until one of the three expires' - AG Macdonell. The magazine is published around the 26th of the month preceding the cover date. There is no January issue. To subscribe, please call 044282 4501, or write to The Subscriptions Manager, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We can invoice your company or take ACCESS and VISA cards. In case of a query about your subscription, please call 044282 4501.
Back issues are available at a cost of £3.50 each - call our Chiswick office (081 994 6477) for a list of issues still in print. We also offer a range of magazine binders, disk holders etc - please call for details.

Editorial
Editorial enquiries should be addressed to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We welcome letters, opinions, suggestions and articles from our readers. If you are interested in contributing articles, please write to this office for a copy of our Contributors' Guide.
Information contained in .EXE is believed to be correct. If errors are found, we will endeavour to publish a clarification in the next available issue.
From time to time, we offer to copy the PC program code described in an article onto diskette for our readers. In this case, please send a blank, formatted disk with a self-addressed, prepaid mailer to the editorial address given above. We can copy both 5.25" and 3.5" disks.
The publishers can accept no liability for any consequences of using software distributed in this way.

Advertising
If you are interested in advertising in this magazine, please write to the Display Advertising Manager/Recruitment Sales Manager (as appropriate), .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH, or call 081 994 6477 for details of our advertising rates.

Pronunciation
The name of .EXE Magazine is pronounced to rhyme with 'not sexy magazine'.

Copyright
Material published in .EXE Magazine is copyright © Process Communications Ltd. Articles (or parts of articles) may not be copied, distributed or republished without written permission from the publishers.

Issue Theme - Life after DOS

32-BIT PROGRAMMING UNDER WINDOWS 3
How Tony Dodd got his 32-bit Prolog compiler running under the mostly 16-bit Windows 3 environment. 12

MIGHTIER THAN THE MOUSE
The writing is on the LCD display, according to Christopher Smithies, who has been looking at GO's new pen-based operating system. 18

TAKING THE XENIX PLUNGE
Richard Samworth has made the transition from DOS to XENIX. He has catalogued his early experiences for us. 26

FROM BROBDINGNAG TO LILLIPUT: GREETINGS!
DPMI is the latest mystery set of initials to emerge from the jargon factory. Dan O'Brien explains what they mean. 35

PERFORMANCE TUNING PARADOX
Andy Redfern has some tricks up his sleeve to get the best out of the PAL language. 46

A CANADIAN SOLUTION
Watcom's DOS and OS/2 C compiler has a 32-bit big brother. Will Watts has been taking a look. 56

SOAPBOX
John Barber dares to criticise the Programmer's best friend. 2

NEWS
We have two new debuggers for Windows, a Windows BASIC from an unexpected source, and even a few non-Windows stories. 4

LETTERS
The best way to test floating-point handling, and Zortech answers its critics. 10

THE THIRD SIDE
Dave Green says that Prolog is the mad computer language from Hell - but he's an enthusiast really. 62

FROM OUR OWN CORRESPONDENT
There are three shady characters at the end of the Danish alphabet who cause a lot of trouble. Ebbe Sønderby explains. 71

CODE PAGE
Do means mesmerize you and standard deviations make you shudder? Overcome your fear of stats as Robert Loughrey proves we're all Normal. 72

CROSSWORD
More clues from Eric Deeson, plus the answer to the March puzzle. 79

UNIX REGULAR
Peter Collinson takes a look at SoftPC: a package which reduces his Sun to a PC. 80

BOOKS
Undocumented DOS and very Documented UNIX are the subjects of this month's books. 88

STOB
Ms Stob has difficulties with a new piece of software. 96

Down with MAKE!

The MAKE tool takes some of the pain out of recompilation. But it should do rather more, argues John Barber.

What sort of a man am I, you ask, to be so unkind? Surely MAKE is every programmer's best friend? Maybe. But it seems to me that MAKE and other utilities which we use for software management do specific tasks well, but they don't reflect a realistic view of the program building process.

MAKE is sometimes criticised for allowing errors. When coding the make file for a large project with many include files, it's all too easy to omit a dependency. This, no doubt, prompted a recent comment that writing a make file for C++ was becoming more demanding than writing the source code. But this aspect is being addressed by the MAKE makers, who now provide Automatic Dependency Generators which do the hard slog of looking for include files.

My criticism of MAKE is at a more fundamental level. MAKE has been busy building programs for years, yet it still doesn't understand its own business. If you are sceptical about this statement, come with me and share my thoughts as I attempt to use MAKE for one of the simplest tasks imaginable: compiling a source file, without include files, to produce an object file.

* * *

'There. I've written a make file just as it said in the instructions. There are two lines. The first one says that my object file has one dependency:. my source file. The second line is a compile command which tells MAKE how to compile the source file if the object file is out-of-date. I'll try running it now ... yes, it's worked, and the object file has been created. I'll run it again to be sure ... it says my object file is already up-to-date, so it hasn't recompiled the source file. Splendid! That's just what I expected.'

'I've just remembered that I should have used a different compiler switch. I'll quickly edit the make file and run it again and...

What's this? It says the object file is up-to-date, it jolly (or words to that effect) - well isn't!'

'I will have to use TOUCH to change the time stamp of the source file. I know this will work, but it seems a bit like cheating, and I always forget. It's lucky that I was concentrating, or I might not have noticed that it hadn't compiled.'

'Perhaps I could be more clever. I'll include the name of the make file itself as a dependency. Then if I change the compiler switches again, I can be sure that the source file will be recompiled. When I add more modules to the make file I'll do the same again ... No, that's no good. If I change the make file, MAKE will think all the files are out-of-date, even the ones that aren't affected!'

'But wait ... I've really cracked it this time. I'm going to put every compile command in a batch file of its own, and call the batch files from the make file. Then I'll declare each batch file to be a dependency of...'

* * *

Using MAKE for such a common task shouldn't be so difficult, so what's going wrong? MAKE is crippled by its simplistic view that the only dependency is the input file. In real life, there will almost always be further dependencies such as the compiler options in the example above. To make matters worse, these dependencies occur in the commands of the make file itself.

It has to be admitted that programmers cope successfully with these difficulties every day, and most wouldn't dream of criticising MAKE; instead they are grateful for its existence. But the fact that MAKE is better than nothing shouldn't blind us to its limitations. Perhaps it is time to lay it to rest.

It's not difficult to see how a better MAKE might be implemented. What is needed is a mechanism for individually time stamping the contents of the make file - it shouldn't be too difficult.

I would like to suggest, though, that there is a wider issue. I used to think MAKE was wonderful. Why? Simply because that was what I had been told and what I had read. It took some time before my mind reluctantly accepted that MAKE was only a crude approximation of what was really needed. I have a horrible suspicion that the same is true of a great many other tools and methodologies which are regularly used by programmers.

EXE

John Barber is a freelance programmer specialising in graphics and user interfaces. Previously, he worked in the systems division of a large engineering company.

THE C LANGUAGE

Microsoft C V6 is a complete rewrite with improved optimisation and a new Programmer's Workbench

C COMPILERS

Zortech C++ OS/2 Dev	OS/2	£360
Lattice C v6	OS/2&MS-DOS	£175
Microsoft C v6	OS/2&PC-DOS	£269
Topspeed C for OS/2	OS/2	£335
Watcom C v8 Prof	OS/2&DOS	£285
Avocet AVCOM 86C	PC-DOS	£210
Aztec C86 Professional	MS-DOS	£120
Aztec C86 Developer	MS-DOS	£175
Aztec C86 Commercial	MS-DOS	£250
Borland C++	Windows/PC-DOS	£250
CI C86 Plus v1.3	MS-DOS	£290
ECO-C88 V4.0	MS-DOS	£ 70
HIGH C v1.61	MS-DOS	£340
HIGH C 386 v2.3	PL386&MS-DOS	£570
Hi-Tech C	MS-DOS	£135
Let's C v4	MS-DOS	£ 60
Microsoft QuickC v2.5	PC-DOS	£ 60
M'soft QuickC/QuickAsm	PC-DOS	£105
MIX C	MS-DOS	£ 20
MIX Power C & lib s'ce	PC-DOS	£ 29
Turbo C++	PC-DOS	£120
Topspeed C Standard	PC-DOS	£135
Topspeed C Extended	PC-DOS	£270
Watcom C v8 std	MS-DOS	£230
Watcom C/386 std	PL386&MS-DOS	£515
Watcom C/386 Prof	PL386&MS-DOS	£680
Zortech C++ v2.1	PC-DOS	£120
Zortech C++ 386 Dev	PL386&DOS	£540
Aztec C86 Developer	CP/M-86	£245
Hi-Tech C	CP/M-86	£150
Lattice C	CDOS	£380
Avocet C	CP/M-80	£210
Aztec C Personal 1.06D	CP/M-80	£120
Aztec Commercial 1.06D	CP/M-80	£200
Hi-Tech C	Z80 + CP/M-80	£ 99
Mix C	Z80 + CP/M-80	£ 35
Aztec C65 V1.05	Apple DOS	£200
Aztec C68K MPW C	MACINTOSH	£110
Aztec C68K C	MACINTOSH	£ 75
Aztec C68K/AM Developer	AMIGA	£175
Aztec C68K/AM Prof'snl	AMIGA	£120
Aztec C68K/ST Developer	ATARI	£175
Aztec C68K/ST Prof'snl	ATARI	£120
Hi-Tech C	ATARI	£ 99
Mark Williams C v2	ATARI	£110
Laser C	ATARI	£135
Prospero C	ATARI	£ 80

C CROSS COMPILERS

We supply 2500AD, Avocet, Aztec, Lattice, IAR and Hi-Tech Cross Compilers hosted on MS-DOS and targeted on Z80, 6502, 6801, 68HC11, 6301, 6809, 7811, 8051, 8096, 68000 & 68020.
Please call for information or advice.

C INTERPRETERS

Instant C/16M supports large programs on 286/386 machines.

C-terp V3.0X	PC-DOS	£165
Instant C/16M	PC-DOS	£700
Interactive C	PC-DOS	£195
Introducing C	PC-DOS	£ 85
Living C Plus	PC-DOS	£135
RUN/C Professional	MS-DOS	£110

MODULA -2 COMPILERS

Logitech Modula-2	OS/2	£400
Topspeed Mod-2 v2 OS/2	OS/2	£115
Stony Brook Prof Mod2	OS/2&DOS	£280
FTL Modula-2 (sml.mem)	MS-DOS	£ 45
FTL Modula-2 (lge.mem)	MS-DOS	£ 55
FTL Modula-2 Developer	MS-DOS	£ 85
Logitech Mod-2/86 Devsys	PC-DOS	£225
Power System w Mod-2	PC-DOS	£120
Stony Brook QuickMod	PC-DOS	£ 95
Taylor Modula-2 Dev Sys	PC-DOS	£505
Topspeed Modula-2 std.	PC-DOS	£140
Topspeed Mod-2 extended	PC-DOS	£270
FTL Modula-2	Z80/CP/M-80	£ 45
Modula-2 System	Z80/CP/M-80	£100
Modula-2 (Mod S/W)	ATARI 520ST	£ 75

Library source is available with some compilers. Please enquire about other libraries and utilities available.

C LIBRARIES & TOOLS

COMMS LIBRARIES

C Asynch Manager V3s'ce	PC-DOS	£120
Essential Comms(L,MS,T)	PC-DOS	£165
Greenleaf Comms(source)	PC-DOS	£175
Multi-Comm (L,MS)	PC-DOS	£190
Silvercomm C Asynch	PC-DOS	£145

DATABASE/FILEHANDLING

Btrieve for OS/2	OS/2	£330
Btrieve for Windows	MS-DOS	£330
Btrieve v5.1	MS-DOS	£330
XQL	OS/2,PC-DOS	£440
Xtrieve Plus	OS/2,PC-DOS	£330
C/Database T'lchest sce	PC-DOS	£ 39
CBTREE	(source any C)	£135
C-Index/Plus	(source any C)	£175
C-ISAM (L,MS)	MS-DOS	£215
C-tree	(source any C)	£265
C-tree Plus	PC-DOS	£315
Essential Btree (s'ce)	PC-DOS	£120
CQL s'ce	PC-DOS	£270
Faircom Toolbox Prof v2	PC-DOS	£645
Pro-C & Workbench v2	PC-DOS	£585
R-tree	MS-DOS	£140
db-VISTA DBMS Object	PC-DOS	£450
Lattice dBCIII (L,MS)	MS-DOS	£185
Lattice dBCIII+	OS/2&MS-DOS	£365
SftFcus Btree&Isam (sce any C)		£ 80

GRAPHICS

Essential Graphics v3	PC-DOS	£200
GFX Graphics	PC-DOS	£ 80
GraphiC v5 (CI,L,DS,MS)	PC-DOS	£290
graphics MENU/META v3	PC-DOS	£125
GSS Kernel system	OS2,PC-DOS	£425
GSS Graph Dev T'kit	OS2,PC-DOS	£425
HALO v3 (MSC5,LAT,TC)	PC-DOS	£200
HALO Professional (C)	PC-DOS	£335
HGraph (MSC)	PC-DOS	£ 80
Ingraf (MSC)	PC-DOS	£185
MetaWINDOW v3.7b	PC-DOS	£150
MetaWINDOW/Plus	PC-DOS	£200
TurboWINDOW/C (Turbo)	PC-DOS	£ 85
QuickWINDOW/C (QC)	PC-DOS	£ 85

SCREEN & WINDOWS

C-Scape +Look&Feel	OS/2&PC-DOS	£345
Vitamin C for OS/2	OS/2&DOS	£240
C View Manager (s'ce)	PC-DOS	£245
Curses/PC w. s'ce	PC-DOS	£240
Entelekon Windows(s'ce)	PC-DOS	£105
Greenleaf Datawindow	PC-DOS	£210
Greenleaf Makeform	PC-DOS	£ 70
Lattice Curses	PC-DOS	£105
Multi-windows (MS,L)	PC-DOS	£190
Panel Plus II (source)	PC-DOS	£270
Power Screen (MS&TC)	PC-DOS	£105
Vermont Views (MSC,TC)	PC-DOS	£345
Vitamin C (source)	PC-DOS	£145

GENERAL FUNCTIONS

C/Utilities T'chest sce	PC-DOS	£ 39
C-Worthy v1.2 + forms	MS-DOS	£340
C Tools Plus/6.0 (MS)	PC-DOS	£ 95
CodeRunner	PC-DOS	£ 95
Entelekon Funct(source)	PC-DOS	£105
Essential C Utils v5	PC-DOS	£145
G'leaf Fin'cial Mathlib	PC-DOS	£210
G'leaf Super Functions	PC-DOS	£160
Greenleaf Functions s'ce	PC-DOS	£125
Lattice C Comp Kit+s'ce	PC-DOS	£160
PDK£1	PC-DOS	£ 75
Power Search	PC-DOS	£ 95
Resident-C (MS,L)(s'ce)	PC-DOS	£145
Silverware C EMM Lib	PC-DOS	£115
The Heap Expander(MS&TC)	PC-DOS	£ 65
Turbo C Tools v2	PC-DOS	£ 95
WKS Library	PC-DOS	£160

MATHEMATICAL LIBRARIES

C Lang Sci Lib v2	ANY C	£160
Mathpak 87 (L,MS)	MS-DOS	£ 80
Science & Eng Tools	MS-DOS	£ 59

TOOLS & SUNDRIES

PC-Lint	OS/2,MS-DOS	£ 85
C-Documentor	PC-DOS	£195
C-Scan	PC-DOS	£195
Clear+ for C	PC-DOS	£140
Databoss (C)	PC-DOS	£320
Lattice Comp Companion	MS-DOS	£ 65

REAL TIME & MULTI-TASKING TOOLS

Concurrent C (PC/MPX)sce	MS-DOS	£ 60
Multi-C (L,MS,CI,TC)	PC-DOS	£190
Op.Sys.T'box (MS) s'ce	PC-DOS	£ 69
Timeslicer v6 (MS6)	PC-DOS	£200
Over-C (L,MS)	PC-DOS	£225

PROGRAMMING TOOLS

Ada Compilers	Algol Compilers
Assemblers & Libs	AWK
Basic Compilers	Basic Interpreters
Basic Utilities	Basic Libraries
BCPL Compilers	C Compilers
C Interpreters	C Libraries
C Utilities	Cobol Compilers
Comms.Libraries	Cross Assemblers
Database Libs.	Debuggers
Dis-assemblers	Editors
Engineers Libs.	Expert Systems
Forth	Fortran Compilers
Fortran Libraries	Graphics Libraries
Icon	Linkers/Locaters
Lisp	Modula-2
Nial Interpreters	OPS 5
Pascal Compilers	Pascal Libraries
Prolog	Rexx
Risc	Screen Libraries
Smalltalk	Snobol

We stock many items for which there is no space in these advertisements.

C++ COMPILERS

Guidelines C++	PC-DOS	£370
Turbo C++	PC-DOS	£120
Borland C++	PC-DOS	£250
Zortech C++	PC-DOS	£120
Zortech C++ DOS Dev	PC-DOS	£270
Zortech C++ OS/2 Dev Ed	OS/2	£360
Zortech C++ 386 Dev Ed	PC-DOS	£540

LIBRARIES & SUNDRIES

C++/Views (ZC++)	PC-DOS	£270
Greenleaf Comms (T&ZC++)	PC-DOS	£195
Object Vision	PC-DOS	£325
Zinc Interface Lib (C++)	PC-DOS	£150
Zortech C++ Database	PC-DOS	£180

DISK COPYING SERVICE

We can copy files to and from 600 disk formats including CP/M, CP/M-86, MS- DOS, PC-DOS, ISIS, APPLE, SIRIUS, BBC, TORCH, APRICOT, HP-150, TRSDOS, DEC RT- 11, IBM BEF, ATARI ST, AMSTRAD, MACINTOSH.

Our charge is £10.00 + disk + VAT with discounts on small quantities and disks are normally despatched within 24hrs of receipt.
For more information call us.

DEBUGGING TOOLS

Multiscope for OS/2	OS/2	£275
Multiscope for DOS	PC-DOS	£115
Periscope I/512K	PC-DOS	£580
Periscope II	PC-DOS	£125
Periscope II-X	PC-DOS	£ 95
Quaid Analyser	PC-DOS	£130
Soft-Ice	386&PC-DOS	£295
Soft-Ice + Magic CV	386&PC-DOS	£400
Turbo Debugger v2	PC-DOS	£105
Zortech C++ Video Course	VHS	£270

CROSS ASSEMBLERS

We supply cross-assemblers by Avocet, 2500AD,Crash Barrier (METAi),IAR Systems and Pecan hosted on MS-DOS, CP/M-86 and CP/M-80 with over 30 target processors. In total over 200 products with no space to list them here. We hold some stock but you should allow 10-14 days for delivery.
Please call for information or advice.

PRICES & DELIVERY

Prices do not include VAT or other local taxes but do include delivery in UK and Europe. Please check prices at time of order, ads are prepared some weeks before publication.

This page lists some of our products. Call us for a complete pricelist.

Order by phone with your credit card.

GREY MATTER

2 Prigg Meadow, Ashburton, Devon TQ13 7DF

TEL. (0364) 53499

GREY MATTER

2 Prigg Meadow, Ashburton, Devon TQ13 7DF

TEL. (0364) 53499

GREY MATTER

2 Prigg Meadow, Ashburton, Devon TQ13 7DF

TEL. (0364) 53499

CIRCLE NO. 646

dbVista V3.2

Raima has announced dbVista V3.2, dbQuery V2.2 and dbRevise V1.03. The new versions have a number of improvements: Raima claims dbVista 3.2 for Windows is twice at fast at writing, and ten times as fast at retrieval as previous versions. dbVista now supports Zortech's C++ V2.16. The company is also working on a set of class libraries for the same. Raima is distributed in the UK by SystemStar, who is on 0992 500919.

Nostalgia corner

DRA has recently obtained the rights to many of the old MicroPro products, which it is now marketing in this country. Included in the 'classic' products are WordStar 3.X (£79), WordStar 4 (£125), SuperSort (£125) and InfoStar (£199). DRA is on 0789 400681.

New Virus Hunter package

Defiant Systems have just announced V3 of their Virus Hunter package. The package has been enhanced in a number of ways, including extra specific virus detection, and the addition of an 'In Depth' mode, which claims the ability to deal with the sinister new stealth viruses, which conceal themselves by emulating low-level BIOS calls.

The package has also updated its resident auditing system, which has now been reduced from a 35 KB TSR to a 400 byte device driver. The package recently won Tandon User Group's Special Program of the Year award. Defiant is on 0752 265777.

DR DOS upgrade offer

A special offer from Digital Research : hand in your old copy of DOS plus £59 pounds, and get a nice new copy of DR DOS 5 back. Phone DR on 0635 35304 for details. The DR Upgrade Program will run until 31st of May.

Real time exec for 386

BSO/Tasking's Real Time Executive now supports the Intel 386 in real or protected mode. The system is fully pre-emptive, and libraries allowing direct interface with Microsoft and Turbo C are also available. BSO/Tasking is on 0252 510014.

Profiler for C

Solution Systems has announced the availability of Charge, a profiler for Microsoft C 5.1 onwards, QuickBasic, Turbo C and C++, Turbo Pascal, MASM and TASM. The system directly interfaces to Brief, allowing programmers to switch between the profile display and the source code. It retails for £79. Solution Systems is on 0763 244141.

Windows Developers

The Windows Development Association is a new meeting group for the UK Windows development community, which will be organising conferences and special interest meetings. An extensive 'Windows bookshop', WDA newsletter and a bulletin board service are also being offered. Membership costs £95.00 per annum for an individual membership, £380 for corporate.

The first conferences were on the 12th of March; one was on programming for Windows, and another on support. Topics addressed included memory management, compatibility with future releases, and DDE. More are coming up. The conferences are a half-day long, and cost £113 for members, £160 for non-members. The WDA are on 0285 640181.

Multiuser DOS from DR

Digital Research has announced a new multi-user, multi-tasking DOS which can support up to 64 users, each using up to eight tasks. Either terminals or PCs can be attached to the 80386/486 system running the operating system. The product is an amalgam of knowledge gleaned from DR's previous DOS products. The task-switching was taken from Concurrent DOS, and the product boasts the same levels of MS-DOS compatibility as DR DOS 5. Windows 3.0 will operate as a task. The filing system has been extended to include user and group file management, and there are also a number of security features, including login passwords, secure disks and hardware keys. The recommended retail price for Multiuser DOS is £495. Call DR on 0635 35304 for further information.

Integrated LISP

LISTPACK+ is a new implementation of LISP for MS-DOS, produced by Mirabilis software. It comes complete with a fully integrated development environment that includes a full screen structure editor, with LISP interpreter and compiler built in. The IDE looks a little primitive compared to other languages, but is certainly better than most PC LISP systems, with screen editing, error linking, automatic indentation and bracket matching, saving, printing and context-sensitive help. It looks and feels a little like early Borland products. The 'compiled' code runs under a LISP virtual machine interpreter: Mirabilis claims 3000 function calls or over 8000 CONS per second on a 12MHz 80286. While the package does not support low-level calls, there is a window interface included, with calls to support viewports onto virtual text screens, menus and forms. LISTPACK+ includes a distributable .EXE virtual machine and manual, and costs £79.50. For more information, phone Mirabilis on 0256 29839.

PostScript for Clipper

Scripton is a linkable library that allows Clipper programmers to lay out documents, select fonts and produce a number of graphics and special effects on a PostScript printer. The library works by returning a string of PostScript commands, allowing the programmer control over whether they should be output immediately, or stored in a file. For example:

```
set print on
? setpage()
? moveto(300,400)
? cshow("Hello")
? printpage()
set print off
```

Text oriented functions in Scripton allow the pen to be moved in conventional printer mode. Other functions draw lines, boxes, arcs and circles. The string returning structure also allows PostScript buffs to tweak the output. Scripton costs £155, and is available from QBS on (081) 994 4842.

Microsoft Launch New Windows Tools

Microsoft has unveiled two new tools for the Microsoft C Professional Development System V6.0. One is a new version of CodeView for Windows which allows single monitor debugging. The other is an extended memory compiler, which supports (but doesn't include) both DOS protected mode interface (DPMI) and VCPI DOS extenders.

The single monitor debugger works via a device driver, allowing programmers to switch between a text mode CodeView screen and the Windows desktop. The minimum system requirements for single monitor debugging are an 8MHz 80286-based PC or better, running DOS and equipped with at least 550K of extended memory, VGA and 5 MB of hard disk space.

The updates are free of charge to all users who purchase the Microsoft C maintenance update, V6.00a. Older registrees should contact Microsoft. The single monitor version of CodeView for Windows is also available from Compuserve in Microsoft's ACCESS area, as well as on our latest .EXE disk (together with the OLE documentation and other Windows developers' tools, such as Michael Geary's FIXDS utility).

Rainbow Technologies -
Leading the world in software protection

Software Developers

Protect your sales and profits with Rainbow's Software Sentinel keys...

- **Sentinel Pro™**
  Rainbow's flagship key uses a variable algorithm technique for maximum security.
- **Sentinel C™**
  Protects multiple packages using developer programmable memory.
- **Sentinel Scribe™**
  Version of Sentinel C which can be dynamically alterable in the field.
- **Sentinel Shell™**
  Protects without requiring access to the source code.
- **Net Sentinel™**
  Site licence revenue protection of LAN applications (NetPro & Net-C).
- **EvE™**
  For Apple Macintosh SE and II, protects Multiple packages, uses ADB port.

Trade marks where used are acknowledged.

With more than 1.5 million Rainbow keys protecting over $2 billion worth of software and a customer base of over 5,000 developers. Rainbow has gained more experience in Execution control, Revenue protection and Information security than any other company.

Rainbow keys offer virtually unbreakable security against unauthorised software use. Custom designed using VLSI and ASIC technology, we have a whole family of Hardware keys, with over 100 high-level language interfaces available for most operating environments, including DOS, OS/2 Windows, Unix, Xenix, Novell and Apple Macintosh.

Worried about customer acceptance?

Rainbow keys are easy to install and totally transparent in operation. Unlimited backup copies can be made (the application will only run when the key is present in the parallel port). A recent survey by Chris Naylor in *PC Week* said *"Rainbow's Sentinel was so easy to set up and to use that compared to some models, it almost felt like cheating"*

We are not the cheapest...

You get what you pay for. We supply a very high quality product, engineered to exacting standards of quality and reliability and backed by the very best in World-class customer support. So when you buy Rainbow, you are not simply buying the latest technology but the support of a team of experts unrivaled anywhere in the World.

That is why many of the best known names in Software Development use Rainbow Keys...

Rainbow Technologies spends far more on Research and Development than any of our competitors. We believe in offering a tried and tested, proven product, which does what it was designed to do, reliably and at a fair price.

RAINBOW TECHNOLOGIES

Call 0753 41512 today for your Free 28 day trial

or fill in the coupon and post it to Rainbow Technologies Ltd., Shirley Lodge , 470 London Road, Slough SL3 8QY

Name .....

Title .....

Company.....

Address .....

.....

Post Code .....

Tel: .....

Yes, I would like the opportunity to evaluate a Rainbow Security Key. I am interested in: (tick box)

- ❑ Sentinel Pro™
- ❑ Sentinel Shell™
- ❑ Sentinel C™
- ❑ Net Sentinel™
- ❑ Sentinel Scribe™
- ❑ EvE™

Rainbow Technologies reserves the right to refuse participation in this promotion

EXE 4/91

CIRCLE NO. 647

WIN A TRIP TO CALIFORNIA!

with BORLAND and .EXE magazine

Borland, the leader in object-oriented programming for DOS and Windows, is to host its first International Languages Conference in San Francisco's historic Sheraton Palace Hotel from April 28 to May 1, 1991 – and you could be there!

The conference has got to be the most important language event of the year.

Designed for software developers and technical managers, the conference will feature presentations from industry leaders on timely topics such as object-oriented programming, developing applications for windowing environments and increasing programmer productivity.

YOU could be joining over 1,000 attendees from all over the world, as well as discovering the immense benefits to be gained from using Borland's acclaimed new language products, **Turbo Pascal for Windows** and **Borland C++.**

How do I get there?

Simply answer the 4 questions below correctly and return the coupon. If you are one of the first TWO names out of the hat, you'll receive an all expenses paid trip to the Conference. Plus a free copy of **Turbo Pascal for Windows** AND **Borland C++.**

And there's more...

As second prizes, we're giving away FIVE free places (worth £295 each) on the London leg of the **Borland OOP World Tour**, Tuesday, May 21st, at the QE2 Centre, London. This is a major international seminar where the world's OOP experts will be sharing their expertise, and paying particular attention to OOP programming for Windows.

Second prize winners will also receive free copies of either **Borland C++** (the only complete C and C++ programming environment for DOS and Windows applications) or **Borland's Turbo Pascal for Windows.** Borland C++ comes with a complete set of tools including a Windows debugger, resource editor and compiler, and WINDOWS.H. – no SDK required! Turbo Pascal for Windows, the world's standard Pascal compiler, is the fastest way to develop Windows applications in Windows.

10 runners up will receive a copy of Turbo Pascal for Windows OR Borland C++ and a further 15 runners-up will receive a .EXE T-shirt of their choice and .EXE mug.

All entrants will receive a free Borland Language Information Pack.

Competition Questions

Q1: Who is Borland's CEO, President and Chairman?
☐ Bill Gates ☐ John Sculley ☐ Philippe Kahn

Q2: What was Borland's very first product (launched in 1983)?
☐ Lotus 1-2-3 ☐ dBase ☐ Turbo Pascal

Q3: Who, or what, is Rikke Helms?
☐ Madonna's real name ☐ An add-on library for Turbo Pascal 6.0 ☐ Borland UK's Managing Director

Q4: What programming magazine's name rhymes with "not sexy"?
☐ TV Times ☐ Viz Comic ☐ .EXE

Winners will be drawn from correct entries received by MONDAY, APRIL 15, 1991. Flight winners will be notified by fax/post the same day. All other prize winners will be notified by post. The judges' decision is final and no correspondence will be entered into. Cut the coupon and send your answers NOW to: Borland/.EXE California competition, Borland International, FREEPOST RG1571, Twyford, Berkshire RG10 8BR: **NO STAMP IS REQUIRED.**

PLEASE USE BLOCK CAPITALS

Name _____

Job Title _____

Organisation _____

Address _____

Telephone _____ Fax _____

If I'm a runner-up, I'd like to receive:
☐ Turbo Pascal for Windows ☐ Borland C++ (Tick one only)

Primary language used _____ Operating System _____

BORLAND SOFTWARE KNOW HOW

CIRCLE NO. 648

Multiscope Windows Emerging

A genuine Windows debugger - that is, one which actually works within a windows and not on a second monitor or text screen - is proving elusive. It isn't here yet. Those baiting their breath for the full Multiscope for DOS/Windows package (originally pre-announced in January) should continue to turn purple. Multiscope has, however, been demonstrating gamma copies of the run-time portion, and it does look impressive. The debugger is fully WIMPed, with separate, iconisable windows for all the main features. The data display and breakpointing design seems particularly impressive. The graphic data window shows structures in text-book format, with boxes, lines and arrows representing records, pointers and values. Breakpoint conditions can be expressed in the host language, and may include statements and even function calls. This allows new code to be attached to points in the program - quick fixes without recompilation. Multiscope has also adopted Superbase 4-like VCR controls, to simplify the control interface.

The launch has been delayed until June while Multiscope sort out the other two thirds of the package: a post-crash module, and a remote debugger (via serial port and network). MultiScope in the USA is, however, arranging an Advantage program, whereby buyers receive immediate shipment of the run-time debugger and documentation, with the rest following as they become available. QA Training markets MultiScope in this country - phone them on 0285 655888 for the latest information.

HP and Sun propose

Sun Microsystems and Hewlett-Packard have submitted a specification for a distributed Object Management Facility (OMF) to the Object Management Group, that would allow sophisticated information sharing over a network or between computers from different vendors.

OMF is based on the NewWave protocols, and is a best described as a high-level system interface - high enough to cope with differences between Sun's ONC and HP's NCS networking systems, but low enough to allow other object managers, such as OLE, to utilise it. The co-operative venture is all the more important as Sun is a big fan of Unix International, while HP is a founding member of the Open Software Foundation. Distributed OMF would also link DOS and Unix systems, and will allow developers to use any of several object-oriented programming languages as well as traditional languages.

Meanwhile, the Object Management Group has announced 12 new members, bringing the membership of the 'Open Systems for Objects' pressure group to 108 companies. The companies are: Cigna, Cinecom, CSX LIS, ERISOFT AB, Genesis Development Corporation, Hal Computer Systems, Hyperdisk, Informix, Qualix, Rational Consulting, Software AG and Stratus. Note that OMG has closed down its British offices, and moved back to Framingham, Mass, so the contact number given last month is now incorrect. Queries, and requests for copies of the Object Management Architecture Guide should now be directed through Eric Leach Marketing, on 081 570 2182. For further information on OMF, call HP on 0344 361428.

CommonView Expands

Glockenspiel has converted the Windows version of its CommonView class library (previously only available for their own C++ cfront preprocessor) for use with the Zortech and Borland C++ compilers. Glockenspiel's President, John Carolan, said that this was in line with his company's strategy of 'making CommonView available for any C++ compiler of suitable technical quality'. CommonView is also making the move to UNIX systems, with development work continuing with Glockenspiel's C++ with Motif and Open Desktop.

The Zortech and Borland libraries cost £399 and are available from QA Training (Tel: 0285 655888). Commonview can be contacted on 010 353 1733166

Digitally Mastered

The full text of .EXE has been available in CD form for some time, as part of the Computer Library service. Ziff, the publisher, has now released a new publication, Computer Select, aimed at purchasers of new products and services. The new package includes 67,000 complete product and vendor information from *Data Sources*, as well as over 60,000 articles and abstracts from 140 leading publications; including us, of course. The publication also contains *The Computer Glossary* by Alan Freedman, and Harry Newton's *Telecom Dictionary*, providing 8,000 in-depth computer industry terms. Computer Select costs $995 for a one-year subscription. Updates are distributed monthly. You'll need a PC, CD-ROM, DOS 3.1 and a High Sierra driver. Ziff's distributor in this country is International Software (081 479 0047).

Watcom p-code compiler
Watcom (0101 519 886 3700) has announced beta testing of its C/p16 compacting and optimising compiler, which uses p-code interpretation on areas of code that are not time critical. An unlimited, royalty free license for C/p16 costs will cost $5,000. The compiler is expected to begin beta testing in March, with general availability planned for June.

Intel loses trademark case
Intel (0793 696000) has lost the first battle in the war to enforce its copyright on 80386 chip manufacturers. Judge William Ingram has announced that the 386 is a generic product designation, and therefore cannot be trademarked. As a result, Intel will now be referring to its chips as the '1386(™)' and 'Intel386(™)' microprocessors. The more controversial issue - whether AMD has the right to copy what Intel claims are products incorporating Intel microcode, is still in the courts.

Marconi piracy case resolved
Marconi Instruments, the company which was served with an Anton-Piller order by the Business Software Alliance some months back, has reached an out-of-court settlement with the organisation. It has also agreed that it will be working with both BSA and FAST (the Federation Against Software Theft) to develop a model control and auditing system for general use.

Help compiler included
One of the criticisms made in last month's Borland C++ article has been addressed: Borland has arranged to license the help compiler from the Microsoft SDK, and this will now be included with the shipped package of Borland C++. Pricing details remain the same: £299.95 retail, £69.95 for Turbo C++ Professional users, £129.95 for Turbo C++ users. Borland is on 0734 320022.

Old Mac transfers
In last month's issue, we mentioned a product which could read Mac disks from a PC, Access Mac, but only if you had a new Mac with the 1.44 MB SuperDrives. Now Peripheral Land Inc has announced a $449 external SuperFloppy for older Macs. The SuperFloppy 1.4 features 94 millisecond average access time, an 8 KB buffer, auto-eject, external SCSI dial, external SCSI termination switch, and an external power supply. Peripheral Land is on 0101 415 657 2211.

Sycero C for Unix
Maidstone-based System C has announced a version of Sycero C, its database program generator, for SCO Unix System V, Release 3.2. The new package is directly compatible with Sycero C DOS source - code should port straight across - and requires no runtime licence. Sycero C Unix sells for £1865 including the code generator and file handler. Existing Sycero C users can purchase Sycero C UNIX for £970 if they own the networking version, or £1270 if they use the single user version. For further information, contact Judy Holly at System C on 0622 691616.

International Standards
Unicode is the new multilingual text and character encoding standard, organised by the larger multinationals. Unicode's backers, who include Apple, Go, IBM, Metaphor, Microsoft, Next, Novell and Sun, want Unicode to be the 16-bit equivalent of ASCII. The draft standard currently incorporates 27,000 separate characters. The group claims that Unicode is arranged to make international character handling as simple and efficient as possible. Now it has a corporation, Unicode Inc, to maintain and promote the standard. It can be contacted on 0101 408 988 8933.

WordPerfect toolkit
WordPerfect (0932 850500) has announced the availability of a toolkit documenting information on screen drivers, file formats, and DOS interrupts on all of the company's DOS based products - including Wordperfect Office's clipboard and hotkeying interface. The package costs £50.

Comsol catalogue out
Computer Solutions' new catalogue is now out. Products covered include the vsDesigner CASE tool; SAGE programmer tools; the Avocet range of compilers and cross assemblers, and a host of FORTH based embedded-system products. The catalogue is available from Computer Solutions on 09323 52744.

dGE upgrade
dGE V4.1, the upgrade to the popular PC database graphics library, is now available from Bits per Second. The new version includes The Graphics Language, which uses Clipper V5.0's User Defined Commands to simplify the calling of dGE functions. dGE will be including TGL support for other xBase compilers that offer UDL like capabilities. dGe V1.4 costs £245; Bits per Second are on 0273 727119.

Interactive gets UNIX V.4

AT&T and Intel have announced that Interactive Systems will be a 'principal publisher' of UNIX V Release 4 on Intel hardware platforms. AT&T has passed on the source, and Intel is transferring its entire UNIX software customer base to Interactive. The hope is that the company, whose V 3.2 is already well established, can bring the new version into wider circulation. The Interactive UNIX will also be iABI-compliant, allowing developers to produce shrink wrapped versions of V 4 applications. Interactive are planning what they call 'aggressive' migration plans to assist all existing and new System V 3.2 users in moving to the new version.

Picking up Pen

Twenty one hardware companies have announced support for Microsoft's Pen extensions to Windows 3. The announcement was made in late February, at the Pen Windows developers' conference held in Seattle. The manufacturers include CalComp, Canon, GRiD, Fujitsu, Hitachi, Kyocera, Sanyo, Sharp and Wang. A limited number of beta copies of the Pen Windows SDK are available from Microsoft at Excel House, 84 Caversham Road, Reading RG1 8LP, or by faxing 0734 596768. Send details about your company and the pen application you intend to develop.

Official C51 User Group

Hitex, one of the distributors of Keil compilers in this country, has broadened the base of its C[80]51 user group. The group was originally formed as a free technical exchange for Hitex's UK registered users, but the company has now been given permission from Keil to turn the group into an international, official user group. Telephone/FAX technical support with guaranteed response times.

Details are available from Hitex's Sarah Batting on 0203 692066. Membership is free for registered Keil users: unregistered users can receive the newsletter for £10, but should phone for an application form first.

BASIC for Windows

Blackwell Scientific Publications, better known for its hardbacks than its software, has announced WinBasic, a BASIC interpreter that runs in and supports Windows. The system is being sold primarily as an introduction to Windows programming, and also as a prototyping system for C programmers.

An interesting point to the product is the manner in which it implements Windows' event-driven nature. This sample program correctly handles mouse, menu and key press messages from windows. All the event dispatching is performed invisibly by the interpreter, rather than by using the traditional ON MOUSE type construction. Some labels are preset, like _KEY and _MOUSE events, while others (like the menu options in the example code) are set up by the user. If a label doesn't exist, WinBasic executes the appropriate default action. BASIC programs can also read and write dBASE files, so you can run those quick file-conversion hacks as a background Windows process. WinBasic sells for £99.50. Blackwell Scientific is on 0865 240201.

```
Dim Menu$(4)

Menu$(1)="Menu1 MenuItem_1001 MenuItem_1002"
Menu$(2)="Menu2"
Menu$(3)="Menu3 MenuItem_3001 MenuItem_3002"
Menu$(4)=""

Menu Menu$

_1001:
_1002:
  a$=Chr$(GetMessage%()-1000+48)
  MessageBox a$+". Item", "Menu1", 0, answer%

_2000:
  MessageBox "Direct working", "Menu2", 0, answer%

_3001:
_3002:
  a$=Chr$(GetMessage%()-3000+48)
  MessageBox a$+". Item", "Menu3", 0, answer%

_KEY:
  a$=Inkey$()
  Print a$, Asc%(a$)

_MOUSE:
  Mouse Button%, x%, y%
  MoveTo x%, y%
  Print "Button "; Button%
```

NDOWS NOW SUPPORTS WINDOWS 3.0 AND MICROSOFT C6.0 NOW SUPPORTS WINDOWS 3.0 AND MICROSOFT C6.0 NOW SUPP

db_VISTA III™

Database Management System

FEATURES	YES	NO
Single and multi-user available	✓	
Relational B-tree indexing	✓	
Network database model	✓	
Multiple database access	✓	
Referential integrity	✓	
Transaction processing	✓	
Automatic recovery	✓	
Record and file locking	✓	
RAM resident		✓
Relational Query and report writer	✓	
Total database redesign/restructuring	✓	
C compilers: most supported	✓	
C ++ compatible	✓	
Operating Systems: VMS, Ultrix,	✓	
UNIX, BSD, SunOs, XENIX, QNX	✓	
MSDOS, MS Windows and Macintosh	✓	
OS/2 compatible	✓	
LANs: Netbios and Appleshare	✓	
Read and write WKS, WK1 and DBF files*	✓	
Source Code available	✓	
Training courses available	✓	
Run-time Royalties (Absolutely NOT)		✓
*using WKS Library		

The db_VISTA III™ database development system is intended for use by the professional C applications developer. db_VISTA III is written in C and provides a complete set of sophisticated development tools that feature:

- High speed access to large or complex data
- Mainframe functionality
- Portability to any C environment
- Royalty-free run-time distribution
- Source code available in C.

db_VISTA III provides both Relational and Network model technology for programming flexibility. Retrieve a record using the relational keyed access method and all related records can be immediately available using the network database model.

db_QUERY provides SQL-like access to **db_VISTA III**, callable from within an application, **db_REVISE** allows restructuring of live databases.

db_VISTA III is fast. Access times for retrieving data are largely independent of the size of the database.

db_VISTA III is portable. Develop an application in any environment and your database access code can be ported without change to any other supported environment. Currently, **db_VISTA III** supports MSDOS, MS Windows, OS/2, Macintosh, QNX, XENIX, UNIX System V and Berkeley, Ultrix and VMS.

db_VISTA III is a product of Raima Corporation and has been sold to thousands of C programmers in over 50 countries. **db_VISTA III** is fully supported in the UK and Eire by Systemstar Ltd. Comprehensive training is available together with bespoke system development. At Systemstar we offer **C-scape 3**™ screen interface library with Look and Feel™ from Oakland Group Inc to provide **db_VISTA III** programmers with a complete application development environment.

For more information about **db_VISTA III, db_QUERY, db_REVISE** and **C-scape 3** call Systemstar in Hertford on **(0992) 500919.**

SYSTEMSTAR LIMITED

1-3 PARLIAMENT SQUARE HERTFORD SG14 1EX
TELEPHONE: (0992) 500919 FACSIMILE: (0992) 554261

CIRCLE NO.

Letters

We welcome short letters on any subject that is relevant to software development. Please write to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. Unless your letter is marked 'Not for Publication', it will be considered for inclusion on this page.

Testing floating point

Sir,

Your March article, 'Software from Hardware' requires comment about programming and computing. You say you borrowed some professional C programming which does Gaussian elimination on a specified matrix. When run in the 32-bit compiler answers could be as much as 4% different from those common to two bit compilers. This is a ridiculous difference. It may indict the 32-bit compiler or, possibly, your Cyrix 387-compatible coprocessor; you will have noticed, in the same issue of .EXE, the comment by Peter Anderton of NAG, 'the results given by some floating-point processors can be dangerously inaccurate.'

Gaussian elimination involves a clearly defined number of arithmetical operations for any specified matrix. Differently compiled programs should be very similar in their machine operations and, therefore, in their accumulated truncation errors. Operations other than the raw arithmetic should contribute little to these errors. If the input matrices you used were large, most of the run time would have been used doing arithmetic so timing differently compiled programs would then give a fair estimate of the compiler efficiency with respect to the simple arithmetic of addition, multiplication and such.

The ridiculous 4% difference in the answers forces one to consider computing rather than programming. Firstly, Gaussian elimination is not the recommended way to invert a matrix. Its variant, Gauss-Jordan, is preferred for reduction of both error and data storage. This method is set out simply in Contribution I/3 of *Linear Algebra* by Wilkinson and Reinsch, where the theory and program are followed by a practical discussion of error. This excellent reference book is used by the computer, the human computer that is, but rarely by the programmer. The test run it gives inverts a section of the Hilbert matrix because we know the exact answer to its inversion. If one uses some random input matrix to invert one has no idea what the correct answer should be using float-point arithmetic. The Hilbert matrix, its inverses and much more are given in *A Collection of Matrices for Testing Computational Algorithms* by Gregory and Karney. If you are to test other 32-bit C compilers on matrix inversion you could save your self a good deal of effort by using the Hilbert matrix as your guinea pig. You could make the arithmetic as lengthy as you wished by increasing the size of the input matrix and you would not even need to input the matrix manually as it is so charmingly simple that a couple of lines of programming would input it for you.

A tougher approach to the error problem would be to do perfect arithmetic using rational numbers in place of decimal ones, but the work...

It is hardly fair to ask you to tackle rational arithmetic but if you want to compare other C compilers for both speed and reliable arithmetic then using traditional test data like the Hilbert matrix is desirable. It would be interesting, also, to see the results with parallel compilations not using the 387 coprocessor.

H Herne
Expert Consultants
Kent

The upside of Zortech

Before I get too depressed by John Cant's review of our C++ Database library *('Zortech and the Database', March '91)*, I would like to do ever such a tiny Baghdad Radio job on his article:

'A great deal of effort has gone into designing a highly valuable toolbox of classes which will be of great value in themselves, and as building blocks for larger structures.

'Zortech has taken a courageous and laudable stand in providing source code for its class libraries.

'Apart from some simple bugs which were quickly fixed by Zortech, I did not find it necessary to delve into the depths of the ISAM file system. The functionality provided was just fine.

'The design of C++ classes is inherently much more complex than that of ordinary library functions.'

On that last point, there is at present a limited population of programmers who are prepared to take the job on. Equally, there is a shortage of programmers willing or able to join in the process of β-testing such products. Obviously, with hindsight, we would have welcomed John on the β-test program.

Also with hindsight we have to admit that the weight of testing during development of the library went into the Btree index and file management classes, including their multi-user capabilities.

The combination of these two factors may account for some of the problems John had with the 'User Interface' of the Database++ library. It is worth remembering that most programmers' function libraries have no user interface whatsoever. Also, the multi-user example presented was intended to be skeletal rather than fully fledged. It is just not possible to pre-judge what use will be made of relatively low-level components of this sort.

At the time the product was issued, all bugs reported by b-testers had been fixed.

Prior to the publication of John's article, all the bugs he mentioned to Zortech had been fixed, and a fixed version sent to him.

As far as I know, at the time of writing all reported database bugs have been fixed, and there is a liberal upgrade policy.

John's remarks about the ISAM file system are particularly welcome. The classes implementing this are the major element of the product, and although it was not particularly visible I get the impression that we got points for those.

Steve Teale
c/o Zortech Ltd
North Yorkshire

Letters submitted to this page may be edited. The writer of the best letter of the month, as judged by the Editor, will be rewarded by a T-shirt or similar-valued .EXE trinket. The best letter is the one printed first.

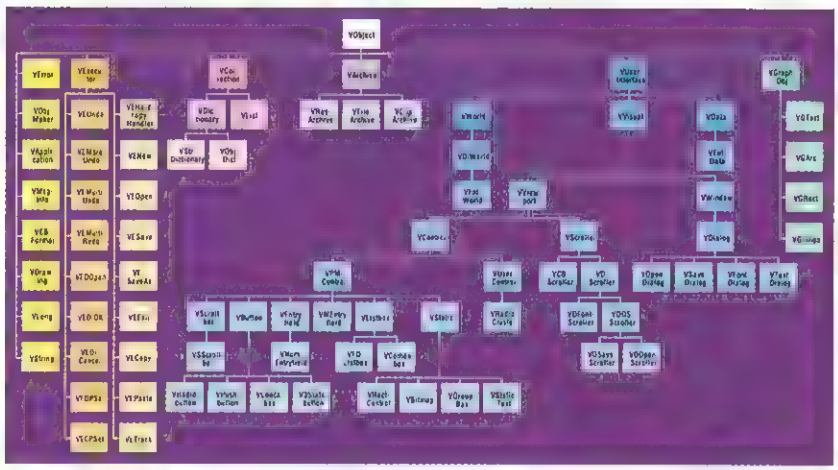
Effectivity in developing MS Windows or OS/2 Presentation Manager programs

If you are building sophisticated C++ applications for MS Windows or OS/2 PM, you could benefit from

- A toolbox with standardized Graphical User Interface (GUI) parts
- An Interface Builder which constructs the standard parts of your program for you
- A Browser so that you can examine and edit the numerous pieces of code of your application

We introduce our *GUI_Master (CLASS-TREE for C++)* which we originally developed for our own use to lessen the burden of writing programs so that one can focus on the application parts and not on the GUI part.

You don't really want to invest your valuable time in reinventing the GUI-wheel?



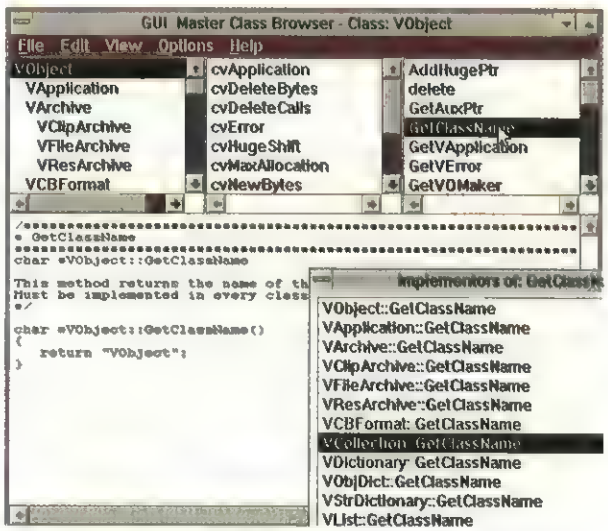
- We don't, so we included over 85 classes containing a wide range of GUI building blocks. Anything that could be accomplished through "normal" programming can be done if you use our product. It's just a lot easier
- Many of the things you might not include in your application because they're too complex or time-consuming to program, are easy to make with *GUI_Master (CLASS-TREE for C++)*
- We provide the Interface Builder, with which you specify all properties of the visual objects of your program. The interface builder then generates the necessary source code, the resource specifications and even the make file
- The Browser enables you to examine your own code and code made by others, and to understand the class structure of that code. From the Browser you can launch the editor of your choice to modify a C++ program

You may want detailed reference documentation

- We supply nearly 1200 pages of detailed class descriptions, method reference etc. An index is included, of course

You may need a cookbook and example programs

- We provide 7 functional example programs and a cookbook explaining the "how to's"



From our experience developers want to use products from someone who knows what he is talking about

- Vleermuis Software Research (VSR) is an independent R&D organization, which, over the past 3 years, spent over 40 person-years on OO development on a broad spectrum of commercial workstations

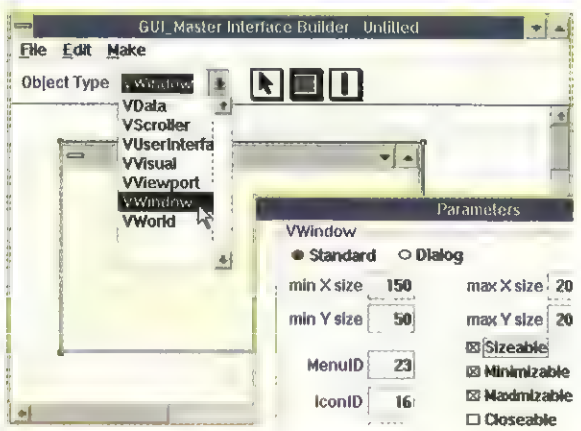
To use our *GUI_Master (CLASS-TREE for C++)* you must have

- A C++2.x compiler, as we do not supply a compiler with our product. We suggest using the Zortech compiler, but we have also tested most of the others. Write for details

We try to keep our prices reasonable

$ 495.= **OS/2 PM version**
$ 545.= **MS WINDOWS 3.0 version**

Prices include airmail delivery (may take three weeks)
Add $ 50.= for courier service
General availability in January 1991 for the OS/2 version
MS Windows version in May 1991



Write or fax your order details to:

Vleermuis Software Research bv
P.O. Box 2584
3500 GN Utrecht
The Netherlands
Fax: Intl. + 31 30 310426

Use Mastercard (number and expiration date) or certified cheque

CIRCLE NO. 650

32-bit programming under Windows 3

When Tony Dodd found that Windows 3 did not offer the 32-bit API he needed to port his Prolog compiler, was he dismayed? Well, yes; but he managed to get round it.

When I discovered the notes tucked away at the back of the SDK Reference for Windows 3 advertising support for 32-bit applications my first thought was 'why hasn't this been accorded more importance?' and my second was 'why is it so small?'. The answer to both questions is the same: Windows 3.0 only supplies functions that you could not write yourself because of the processor's protection mechanism. It doesn't provide functions that you would expect to find in a normal development environment.

However, systems programmers can take these functions and use them to make 32-bit applications accessible to users. This article describes how I ported one such application, Prolog-2, which is an implementation of the Prolog language for PCs. I hope to encourage other developers to change their languages and toolkits so as to provide easy access to the full power of the 80386 for those who cannot cope if only given the bare essentials supplied by Microsoft.

Because this is an article for system developers, and because a small book would be needed to explain all the conventions, standards, architectures and interfaces involved, I shall assume that the reader already understands the different members of the 80386 family.

Protected mode programs

The 80386 processor contains a museum of old PC architectures too complex for consideration here. To summarise very briefly:

- if the processor operates in real mode then it behaves very like an 8086;
- if the processor operates with the virtual 8086 bit set then it behaves rather like an 8086 but uses paging so that the 8086 memory need not be the first 640 KB of physical memory;
- if the code being executed is in a small code segment (a `USE16` segment, as the assembler calls it) then it behaves like 80286 protected mode code.

We have all got used to PCs running in real mode when the operating system is MS-DOS, and protected mode when it is OS/2; but Windows 3 has changed all that. There are three modes:

- in real mode the processor is in real mode;
- in standard mode the processor is in protected mode but switches to real mode to run DOS applications;
- in 386 enhanced mode the processor is in protected mode and DOS applications are run as virtual 8086 tasks.

We shall only be concerned with Windows applications in this article, so we must address the problem of writing protected mode programs. In fact this isn't usually a problem. Compiled C programs should not make assumptions about the values in segment registers, so the program should not care whether segment registers contain physical addresses (as in real mode) or offsets in a table (as in protected mode). A few problems may be envisaged: for example, what about huge arrays? How can the compiler produce portable code to pass a 64 KB boundary? The solution in Windows 3 is particularly ingenious: successive segments that tile an area greater than 64 KB are arranged in sequence in the segment table, so that the distance between them is fixed. In real mode this distance is also fixed - it is 1000H. So in each case a constant needs to be added to a segment register to pass a 64 KB boundary; and this constant is defined appropriately by the Windows kernel.

System services

Suppose we agree, then, that it is not impossible that a Windows application will run in protected mode as well as in real mode. Directly or indirectly the code will almost certainly issue DOS calls through interrupt 21H; should we assume that these will work too?

The answer to this question must be a resounding no. First, DOS operates in real mode, and makes assumptions about segments that prevent it from running in protected mode. Second, any parameters passed to DOS that use segment registers will be interpreted quite wrongly; DOS will convert them to linear addresses by multiplying them by 16, which isn't the idea at all. Third, protected mode interrupts do not use the interrupt locations in low memory; they use a separate interrupt table.

Issuing interrupts in protected mode and hoping that they will work as in real mode is therefore not advisable. However, in order to make Windows applications operate correctly in protected mode, Microsoft provides almost all DOS services via the protected mode interrupt 21H (which, as we shall see later, is a 286 trap gate). The few restrictions are summarised in Figure 1.

It should be noted that providing these services involves more than the translation of a few addresses. For example, to read from a file into a buffer at a protected mode address is straightforward only if the pro-

tected mode address has a real mode equivalent; otherwise some clever remapping of pages is called for. For this reason, more obscure DOS calls cannot cope with transfer of more than 4 KB of data at a time.

A few DPMI calls are of use to the Windows programmer, however. *(For a detailed description of DPMI, please see Dan O'Brien's article elsewhere in this issue - Ed.)* For example, access to an interrupt other than 21H may require direct use of DPMI. My example shows how the DOS case conversion function may be accessed from protected mode; of course, Windows 3 already has a perfectly good case conversion function, but the use of the DOS version serves as a nice simple example.

Recall that DOS call 38H returns, among other things, the address of a procedure that can be called to convert a character to upper case. As this address is only suitable for use in real mode, we must use a special DPMI call to invoke the routine from protected mode. A parameter block containing the desired register settings in real mode is built and passed to DPMI; the code for the example is given in Figure 2.

Of course, code like this must be used with caution; it will not work in real mode. Programs that need to do such low-level operations must use `GetWinFlags` to find out which mode the processor is in.

32-bit programs

32-bit code offers the programmer tremendous advantages. The full address space of the machine can be accessed from a single offset register, without the use of segments. Segments are still useful as a relocation and protection mechanism, but they are no longer needed as a way of accessing large blocks of memory. My own PC software product, Prolog-2, has always been hampered by the segmented architecture; languages like Prolog like to have a large pool of memory available in which objects of any size can be allocated and deallocated at will. Huge arrays are no solution, and most PC Prolog implementations have had to restrict run-time stacks to 64 KB. The 32-bit architecture of the 80386 solves all that.

Unfortunately it is not that simple. Presented with a chip that at last lifted the major limitation of the 8086 family of processors and made available 32-bit facilities comparable to those that had been in every other major processor for years, what did Microsoft and IBM do? They designed a new OS that prevented code from running in 32-bit mode. They forced users to operate their nice new 80386 processors as fas-

Function	Description	Status
00H	Terminate Process	Not available
0FH	Open File with FCB	Not available
10H	Close File with FCB	Not available
14H	Sequential Read	Not available
15H	Sequential Write	Not available
16H	Create File with FCB	Not available
21H	Random Read	Not available
22H	Random Write	Not available
23H	Get File Size	Not available
24H	Set Relative Record	Not available
25H	Set Interrupt Vector	Only affects protected mode vectors
27H	Random Block Read	Not available
28H	Random Block Write	Not available
35H	Get Interrupt Vector	Only affects protected mode vectors
38H	Get Country Data	See text for details of how to call the case mapping function
44H	IOCTL	In functions 2,3,4,5 do not specify more than 4K of data. Function C not supported.
65H	Get Extended Country Information	See 38H

Figure 1 - Limitations of DOS calls in protected mode

```
; This code calls a real mode routine. The
; address to be called is in EBX (real
; segment in top half) and the argument is in
; AX. If [EBX] is the address of the DOS case
; mapping call and AX has a character code
; then AX will return with the converted
; value.
        sub     ESP,34H         ; allocate the call

                                ; block (34H bytes)
        mov     EDI,ESP
        push    SS              ; Call block address
        pop     ES              ; in ES:DI
        push    AX              ; save argument
        push    EDI
        mov     ECX,34H
        mov     AL,0
        rep     stosb           ; zero block
        pop     EDI
        pop     AX
        mov     1CH[EDI],AX     ; set real mode AX..
        mov     EAX,[EBX]       ; .. IP and CS
        mov     2AH[EDI],EAX    ; note: real mode
                                ; stack will be

                                ; supplied as SS=0
        mov     AX,0301H        ; DPMI call real mode
        mov     BH,0            ; reserved flags
        mov     ECX,0           ; leave the stack
                                ; alone
        int     31H             ; DPMI
        mov     AX,1CH[EDI]     ; get answer
        add     ESP,34H         ; clear block
```

Figure 2 - Using DPMI to access a real mode function

```
alloc     dd      ?
selector  dw      ?
...
          push    alloc             ; amount to allocate
          push    DS                ; FAR pointer to selector
          push    OFFSET selector
          push    alloc             ; max size of allocated object
          push    WORD PTR 0        ; reserved
          call    Global32Alloc
```

Figure 3 - Allocating a 32-bit segment

```
          mov     ECX,20000H        ; number of bytes
          mov     EDX,0             ; buffer address
          mov     DS,selector
          mov     AH,3FH
          int     21H
```

Figure 4 - Reading into a 32-bit segment - wrong!

Don't leave your customers HANGING. . .

BEFORE YOU SHIP ANY PROGRAM, CHECK IT WITH

COMPUTER LANGUAGE PRODUCTIVITY AWARD 1990

BOUNDS-CHECKER™

WINNER PC MAGAZINE BEST OF 1990

JANUARY 15, 1991 PAGE 168

FINDS OUT-OF-BOUNDS MEMORY ACCESSES *AUTOMATICALLY*

Your program may have 10,000 to a million lines of code. It may occasionally hang mysteriously or it may appear to run flawlessly every time. But under DOS, how can you ever be sure that your program is not corrupting memory it does not own? The only way to be 100% sure is to BOUNDS-CHECK before you ship.

To use BOUNDS-CHECKER you build your program with debugging information (we support most compilers including Microsoft, Borland & JPI). Then you just type <BC file-name>. BOUNDS-CHECKER sets up the 386™ /i486 for protection and lets your program fly. If your program accesses memory it does not own or overwrites its own code, BOUNDS-CHECKER pops up displaying the offending SOURCE-LINE or instruction.

Programming under DOS is a gamble, so why not stack the odds in your favor--CALL TODAY.

(603) 888-2386

Call by 4:00 PM EST TODAY and ask us to EXPRESS you an info packet. In most cases you will receive it by 10:30 AM tomorrow. (USA only)

OTHER FINE NU-MEGA PRODUCTS. . .

The ultimate systems debugger.

Debug:
- Interrupt routines
- Device drivers
- T&SRs
- ROMS
- Applications
- Overlays

Soft-ICE™

Features:
- Break out of a hung program
- Real time Break-Points
- Back-Trace history
- Works with other debuggers

If you are debugging an application, Soft-ICE is seamlessly integrated with BOUNDS-CHECKER so you can easily go back and forth between BOUNDS-CHECKing and debugging: a combination many programming professionals can't live without.

1988 BYTE AWARD OF DISTINCTION

Run CODEVIEW for Windows on a single monitor
CV/1.....$129
• Runs in a window • No annoying flash •
• Runs on any display that supports windows •

All Nu-Mega products require a 386, 386SX or 486. MS-DOS and Codeview are trademarks of Microsoft Corp. 386 is a registered trademark of Intel Corp. Nu-Mega, BOUNDS-CHECKER, Soft-ICE and CV/1 are trademarks of Nu-Mega Technologies, Inc.

™ Nu-Mega TECHNOLOGIES INC

P.O. Box 7780 • Nashua, NH • 03060-7780 U.S.A.
(603) 888-2386 • Fax (603) 888-2465

BOUNDS-CHECKER $249
Soft-ICE $386
Special Offer
Buy BC & S-ICE .....Save $100

30 Day
Money Back Guarantee

CIRCLE NO. 651

ter versions of the ghastly old 80286, and every few months they made an announcement about a 32-bit version of OS/2 being available at some unspecified future time.

A short term solution was the DOS extender. A DOS extender is a small 32-bit operating system that can be bound with a 32-bit application and run from DOS. The DOS extender switches the processor into protected mode, and offers services comparable to DOS services to the application; it may also allow it to communicate with real mode code. I used the Phar Lap DOS extender for the 80386 version of Prolog-2; it is an excellent system, well tested across a range of odd PCs and very nicely documented. Virtual memory support is also available. But there are problems with DOS extenders:

- a DOS extender application cannot run as a DOS application under Windows 3, though this will be fixed when DOS extenders support DPMI.
- a DOS extender application cannot run as a Windows 3 application, and I know of no plans to fix this.
- DOS extender vendors charge run-time licence fees. Though this would not be too unreasonable if one were selling a packaged application, there is a problem for those selling language compilers that require the run-time support of a DOS extender; people who use the language to ship applications must pay run-time fees to the DOS extender vendor.

The services that a DOS extender offers may be divided into groups:

- it switches the processor between real and protected mode.
- it loads applications.
- it offers operating system services to applications.

We have seen that Windows 3 offers all these services as well; but only for 16-bit applications. On the other hand, whereas moving from real to protected mode requires a processor mode switch, a `USE32` segment is a mere long jump away from a `USE16` segment.

There is one technical difficulty: Windows normally allocates `USE16` segments for code. To load a 32-bit application it is necessary to get a large data segment, read the code into it as though it were data, and then change the segment to a `USE32` code segment. Because Windows 3 applications cannot use privileged instructions, it is impossible to go and hack these segments out of the segment table: but Windows 3 provides a set of functions, grouped in the 32-bit DLL (described in Appendix E of the SDK Reference Manual), that perform just the tasks required.

In summary, then, to run a 32-bit application under Windows 3 we need a loader program that will load the code as described and transfer control to it, and we need to supply the code with system services.

Of course, we also need a 32-bit application. Several compilers and assemblers are available; I use the Phar Lap set. Whichever you use, make sure that you have a description of the linker output format, as you will have to load the code yourself. Make sure that the application is written to use a single segment for all code and data.

Loading a 32-bit application

Loading a .EXE file is not one of life's great pleasures. But apart from the odd page size, most of the annoyance derives from the need to fix up segments. In a flat 32-bit application that problem does not arise: there should be no segments hard-coded into the program. For various odd purposes, like accessing the PSP, my Phar Lap program used fixed integer segment numbers; such things must, of course, be removed, because they assume the particular segment usage of Phar Lap's own DOS extender.

Before loading the program you need to know how much memory to allocate, how much memory to load, where the stack pointer is to go and where the code begins. All these items will be available in the header, though the exact format will differ from linker to linker. In more detail, the sequence of operations is:

- open the file to be loaded.
- read the header and extract the fields you need.
- use `Global32Alloc` to allocate the desired amount of memory.
- read the load image from the file into memory.
- close the file.
- use `Global32CodeAlias` to get a code segment alias for the data segment.
- switch to the application stack.
- perform a long jump to the start address.

Because the loader program needs to mix 16-bit and 32-bit offsets in a complicated way it is easiest to write it in assembler. The loader segment needs to be declared as `USE16`. Remember to use Pascal calling conventions for Windows routines; in other words, push the first argument onto the stack first. Thus the initial call to allocate memory may be as in Figure 3 which leaves the return code in AL and the segment selector in `selector`.

Reading the load image requires some thought. If there are 20000H bytes it will not do to use the code in Figure 4, because DOS will ignore the top half of the ECX register and simply read 0 bytes. The file must be read in portions smaller than 64 KB. So we may start with the code;

```
mov     ECX,0C000H
mov     EDX,0
mov     DS,selector
mov     AH,3FH
int     21H
```

reading the first 0C000H bytes into memory. This works ok, but it should be clear that adding 0C000H to EDX and doing the same again will not work. It will correctly read the 4000H bytes of code starting at DS:0C000H, but at the end of the 64 KB segment DOS will wrap around and start again at DS:0.

Fortunately there is a call available to remedy this: we can obtain a new selector for a segment starting at DS:0C000H, as in Figure 5. Such a selector is called an *alias*, and allows you to tile segment selectors so that the whole of a very large segment is covered by tiles whose individual sizes are less than 64 KB. Now the data may be read using the code;

```
mov     ECX,0C000H
lds     DX,alias16
mov     AX,3FH
int     21H
```

When the operation is finished, the 16-bit alias should then be freed with `Global16PointerFree`.

```
alias16     dd      ?
            ...
            push    selector                ; Segment to be aliased
            push    DWORD PTR 0C000H        ; Offset to be aliased
            push    DS                      ; FAR pointer to alias16
            push    OFFSET alias16
            push    DWORD PTR 0C000H        ; Size of alias segment
            push    WORD PTR 0              ; Reserved
            call    Global16PointerAlloc
```

Figure 5 - Getting a 16-bit alias

```
aliasc      dw    ?
            ...
            push  selector            ; segment to alias
            push  DS                  ; FAR pointer to aliasc
            push  OFFSET aliasc
            push  WORD PTR 0          ; reserved
            call  Global32CodeAlias
```

Figure 6 - Getting a code segment alias

Jumping into the data segment that we have just loaded with code would cause an error: but getting a code segment alias is straightforward: see Figure 6. At this point all that remains is to start the application, as in Figure 7. It is for the sake of tricks like this that we must program in assembler.

Generally assemblers have not caught up with mixed `USE16` and `USE32` segments. The assembler can only cope with `near` and `far` procs. In a `USE16` segment a `near proc` pushes just a return offset on the stack while a `far proc` pushes a 16-bit offset and a segment on the stack; but nothing pushes a 32-bit offset and a segment. Similarly with return statements.

In the instruction encoding of the 80386 processor there are not separate opcodes for 16 and 32 bit operations. Instead, the processor looks for a 32-bit operand in a `USE32` segment and a 16-bit operand in a `USE16` segment; but the code generated is just the same. How, then, can a 32-bit operand be used in a `USE16` segment? The answer is the size override byte, which has no symbolic name but happens to be 66H. We have to define `big`;

```
big   macro
      db        66H
big   endm
```

and insert the override in the code stream ourselves. (Of course, you can just put

```
db    66H
```

in the middle of your program if you insist).

System services

You cannot use int 21H instructions in the 32-bit part of your program and you cannot link it with library routines that contain such instructions. Interrupt 21H in protected mode is a 286 trap gate, which means that when it returns it will pop a 16-bit offset off the stack, and therefore, probably, return somewhere strange, say in the middle of an instruction.

System services must be provided within the 16-bit part of the application. It is, therefore, necessary to allow the 32-bit code to call the 16-bit code (probably you need to call the Windows library too, and every so often yield control to other applications).

It isn't possible to link the `USE32` segment directly with the loader program, for obvious reasons. Instead the `USE32` routine should be replaced by a stub that calls across to the 16-bit using a jump table. Thus, suppose that we have a routine `openf(char*)` that opens a named file. Instead of opening the file directly the `USE32` routine may say something like the code;

```
_openf  proc
        call    [jtable+OPENF*4]
        ret
_openf  endp
```

All that this does is to look in a jump table to find the address of the 16-bit open routine and transfer control there: `OPENF` is a suitable literal. The jump table must be initialised at the start of the `USE32` code based on a table passed by the 16-bit code.

A pointer to the name of the file to be opened is passed on the stack. Provided that the `USE32` program keeps its stack below offset 10000H, the two programs can use the same stack, and I would strongly recommend doing this if possible. If use of a high level language forces you to put the stack higher than this then every call to `USE16` code will have to switch to the 16-bit stack and copy the appropriate number of arguments from the `USE32` stack; and afterwards, switch back.

Another problem will arise if the name of the file is above offset 10000H; this is hard to avoid. In that case you can either copy the name lower (but this gets very tedious in some cases, such as buffered I/O) or you can take a 16-bit alias of the address as we did in reading the load image. Provided you write a general purpose table based interface rather than generating each piece of code separately, this is the better approach.

```
        lss     ESP,appstack
        push    aliasc
        push    appstart      ; 32-bit start address
        big
        ret
```

Figure 7 - A 32-bit call in a USE16 segment

Conclusion

It is hard to estimate the work involved in converting any application. It is a great advantage that you need only convert calls used by your application; you don't have to produce a general purpose interface. I will only say that it took me much less time to convert Prolog-2 than I had expected.

I suppose it would be fair to say that 32-bit programming under Windows 3 is not for the timid. Later releases will make it easier. For the programmer who needs to deliver 32-bit applications on the PC, the Windows 32-bit DLL has unique advantages: it's available now, it works, and you can give it away with your application.

PS

Microsoft has announced that there will be a 32-bit version of the Windows API, called *Win 32*. This will make all the contortions described in this article unnecessary, and it will be possible to link 32-bit code directly with the Windows libraries. Moreover, the next release of Microsoft C will come in two forms: 16-bit or 32-bit. The developer's tools for Win 32 are due to be shipped in late 1991, but there will not be a retail version based on DOS until 1992; moreover, it is expected that many users will stay with 16-bit Windows.

More immediately, the WLO libraries allow Windows applications to run under OS/2. OS/2 1.x will not support WINMEM32.DLL, but OS/2 2.x together with WLO 1.0, which will be part of the SDK for Windows 3.1, will allow WINMEM32 calls. So anyone developing a 32-bit application in the way described in this article should be able to supply a common .EXE file that will work both with Windows and with OS/2. *(We will be printing an in-depth article on WLO in the near future - Ed.)*

Tony Dodd is the Chairman and Managing Director of Expert Systems Ltd. He was the designer of the Prolog-2 implementation of Prolog-2 and has written textbooks on Prolog and on Prolog implementation. He can be contacted on 0865 794474, or via Email at tdo@uk.ac.exeter.cs.

What Prolog programmers have been waiting for !

Prolog-2 for Windows
version 3.0

Once again Expert Systems Ltd. are first to deliver a Prolog system matched to today's programmer's needs.

Our new Prolog-2 for Windows 3 is the only package that allows programmers to combine the power of Prolog logic programming with the acclaimed GUI of Windows 3.

Now **you** can create 32-bit applications for the world's favourite GUI with **your** favourite programming language.

Just look what Prolog-2 for Windows 3 gives you :

- Full 32-bit addressing using 386 enhanced mode, no more limits on workspace, heap etc.
- Virtual memory support
- Startling speed. Up to 150 kLIP on a Research Machines VX486
- Full bidirectional C interface for embeddability
- Integrated window editor and graphics
- Over 500k of Hypertext help
- Workstation performance on a PC for just £1495.00

I need a Prolog to write Windows 3 applications right now. Please rush me Prolog-2 for Windows 3.

Please add £10 p&p plus V.A.T.

My cheque is enclosed
or My official order is attached
or I wish to pay by Mastercard/Visa
All trademarks acknowledged

copies at £1495 (£747.50 each)
Card no.
Expiry Date
Name and address
Signature

Expert Systems Ltd.

Unit 12, 7 West Way, Oxford, OX2 0JB. England. Tel 0865 794474 fax 0865 250270

CIRCLE NO. 652

SEE US AT SOFTWARE TOOLS 91 11th - 13th June STAND 58

System Architect™

A PC based Microsoft Windows CASE tool

Microsoft Windows V 3.0 compatible

Ψ
REAL TIME SOFTWARE

118-120 Warwick Street, Leamington Spa, Warks CV32 4QY

TEL: 0926 450858

CIRCLE NO. 654

"How Did We Ever Manage Without SMS?"

ASM (Europe)

SMS provides a comprehensive set of tools for automating the process of managing different versions of files and configurations of systems. SMS improves *quality, reliability* and *productivity.*

Version 2.1 now available! Even more power and flexibility

Software **M**anagement **S**ystem provides a fully integrated environment; all features can be accessed through a menu-driven Front-end with context-sensitive on-line help.

SMS
SOFTWARE MANAGEMENT SYSTEM
INTASOFT

Version Control
Management of multiple revisions, lines of development and users

- Revision Storage and Retrieval
- Edit Control
- Revision Identification and Reporting
- Minimisation of Storage
- Text and Binary File Support

Configuration Management
Management of configurations of the items which make up a project/product

- Release and Configuration Identification
- Automated System Building
- Variant/Configuration Maintenance
- Automated Dependency Generation

Modification Requests
Formalisation and tracking of bug reporting and upgrade procedures

- Change Management
- User-definable Life-cycle
- Active Notification System
- Modification Analysis and Reporting

Available for: MS-DOS, OS/2, Unix, VMS, OS-9
Single User PC Price £ 490.00 + VAT ; 5 User PC Price £ 980.00 + VAT
Please contact us for further technical and pricing information

INTASOFT : **the productivity tools for professional software developers**

Tresco House, 153 Sweetbrier Lane, Exeter, EX1 3DG, England, Tel: 0392 217670, Fax: 0392 437877

CIRCLE NO. 653

MIGHTIER THAN THE MOUSE

After three years of unparalleled secrecy, GO Corporation has announced an object-oriented OS for a new kind of machine: the pen-based PC. Christopher Smithies has been evaluating the system.

On 22 January 1991, after a period of undercover productivity, GO Corporation unveiled its new operating system: PenPoint. At the launch, GO demonstrated PenPoint on a pen-based machine of its own design and manufacture, and representatives from hardware producers IBM, NCR and GRiD gave brief presentations attesting their commitment to the new system. What, then, does GO offer to attract such interest from third parties?

The User Interface

With the mouse/keyboard interface, it is frequently necessary to issue commands in two steps: first, get to the right place with the mouse, then key the appropriate command.

With a pen, one simply writes the command in the appropriate place. Under PenPoint, commands may be alphanumeric characters (which make use of handwriting recognition) or meaningful gestures: caret for insert, cross-out for delete, square brackets for block-marking, question-mark for help, and so on. A corollary of this is that there is no cursor. Characters and commands are entered wherever the pen is put down. The pen becomes, as it were, a 'random access' device and potentially represents as big a step forward for the user interface as the move from tape to hard disk did for data processing.

The user interface is based on the familiar desktop concept. But whereas the older GUIs do everything with windows, PenPoint also offers a different metaphor: the notebook.

A notebook presents itself as a window almost filling the screen. It is subdivided into pages, each of which roughly corresponds to a disk file. The initial page is a table of contents, presented in much the same way as the table of contents in a printed book - see Figure 2.

A page may be selected by a pen-tap on its name, and may be turned-to by a tap on the page number. As a convenience, the user may add any number of index-tabs to a notebook. This is achieved by writing the letter 'T' over any entry in the table of contents. This causes a tab to appear on the right-hand margin of the notebook window. A single pen-tap on any tab will cause the appropriate page to be turned-to (ie the appropriate document or file to be displayed as a notebook page).

As you can see from Figure 2, the table-of-contents display is self-explanatory. The underlying software is replete with features which are invoked with easily memorised gestures. For example: to create a new document one might make a caret ^ mark (for 'insert') at a particular place on the table of contents. A menu appears offering a range of possible document types, known as 'stationery'. One makes a selection by tapping with the pen upon the appropriate menu item (eg 'Notepaper'). A new page with the name 'Notepaper' appears on the table of contents, inserted in the chosen place. If we were to turn to it, this new page would be blank.

Of course, 'Notepaper' is a generic title. One would probably want to rename the new document, thus giving it a specific identity. This is done using the same modify command as would be used to change a word in a document, or a line in a drawing. You just use the modify gesture: draw a small circle over the word 'Notepaper'. When the modify gesture has been issued, an edit box appears. See Figure 3.

In turning to a page of the notebook, a dormant application is brought to life. Under PenPoint, the operator does nothing specifically to load or activate an application: this is regarded as part of the outmoded disk-based concept. An application is permanently loaded upon installation and a new process is initiated whenever a document is created. This process would normally 'sleep' when the document is not displayed.

All representations of datasets are referred to as 'documents'. Applications are considered not from the point of view of the running process, but rather from the input forms necessary to provide them with data or the output views representing these data. Input forms and output views of data are accessed as pages within the notebook.

- commands issued via pen gestures and handwritten characters
- window-type GUI using notebook metaphor
- handwriting recognition integrated into system
- pre-emptive multi-tasking
- object-oriented design
- instant power-up/power-down support
- can use any host filesystem (currently MS-DOS)
- multiple simultaneous network support, automatic logon/logoff
- deferred input/output
- device-independence
- user-interface components (window layouts, fonts etc.) scalable
- transparent file and directory system
- flexible and powerful inter-process communication

Figure 1 - Salient features of PenPoint

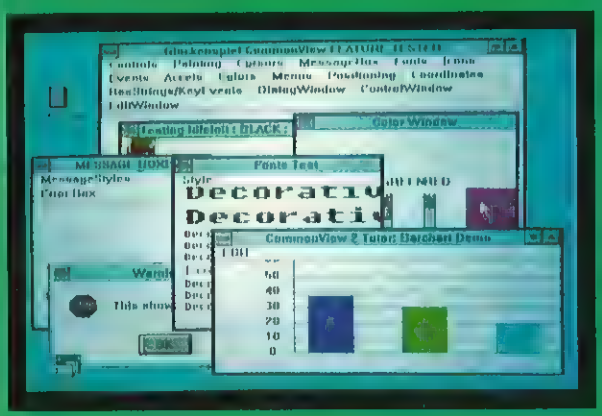
Objects of Desire

glockenspiel CommonView 2

***Glockenspiel CommonView** has really made its mark in the world of Windows development. Thousands of developers have used it to speed up and simplify their projects.*

Now **CommonView 2** is available, inheriting the success of its predecessor and extending its capabilities even further to deliver efficient Windows 3.0 apps.

That's because **CommonView 2** works with **Glockenspiel C++ 2.0**, giving you a C++ object-based framework that reduces the complexity, cuts the code, manages memory and lets you stay in touch with what you're really doing. From compilation to execution, **CommonView 2** applications are fast and powerful.



CommonView 2 objects model the real world of Windows. And that's just the beginning - from there **CommonView 2** takes you as far as you want to go.

It doesn't anticipate everything. That's where C++ takes over: its inheritance capabilities enable you to extend the **CommonView 2** framework. Just take an existing object and customize it without affecting the original. A few additional lines of code and you could, for example, turn an edit control into a password control. Then reuse it in other applications.

You *don't* have to learn a complete new language, you *can* integrate existing developments; it *is* portable to PM and NewWave; CodeView *can* be used on your C++ source.

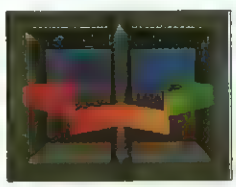
The world wants Windows 3.0 apps now! And with **CommonView 2** you can inherit the earth.

Specifications.

Glockenspiel CommonView 2 includes Glockenspiel C++ 2.0 and Container - the object storage framework. It requires Microsoft C 6.0, the Windows SDK and 1.5 meg of memory. You debug C++ source with Microsoft CodeView 3.0. Glockenspiel C++ supports a completely portable memory management system. Glockenspiel CommonView consists of approximately 65 classes.

Comprehensive documentation includes CommonView tutorial and reference manual, Glockenspiel C++ compiler manual and User Guide, C++ syntax and AT&T Library Guide, pullout guide to compiler switches, *plus* "Programming in C++" by Stephen C. Dewhurst and Kathy T. Stark (Prentice Hall).

On-line hypertext documentation for CommonView reference manual and AT&T guides. Tutorial source code also on disk.



Glockenspiel C++ works from within the Programmer's Workbench with the reference guides instantly available from the on-line Advisor, using Microsoft CodeView for debugging.

Glockenspiel CommonView applications are portable between Windows 2.1 and Windows 3.0, PM 1.1 and PM 1.2 with HP New Wave 3.0 version coming soon.

glockenspiel
class constructors

QA

QA Training Limited,
Cecily Hill Castle,
Cirencester, GL7 2EF.
Phone: 0285 655888. Fax: 0285 650537.

CommonView costs £495 + VAT and is available for Windows 3, OS/2 Presentation Manager and HP NewWave.

Glockenspiel Professional C++ 2.0 Compilers are available for DOS, OS/2 and Workstation platforms. Call for details.

Glockenspiel CommonView and Glockenspiel C++ are registered trademarks of Glockenspiel Ltd. The trademarks of their respective corporations are acknowledged.

CIRCLE NO. 655

A database application might be implemented via several notebook pages; some of these pages might present input forms for the addition of data to the database, while others would present reports (or browsers) on the current contents of the database. Rather than explicitly request a particular report, therefore, the user would turn to the appropriate page in the notebook. The concept of 'groups of pages' is supported via the concept of notebook 'sections', within which sub-documents may be individually selected. This notion directly corresponds to the traditional hierarchical directory model. Any number of notebooks can co-exist on the desktop. The help system, for example, uses a separate notebook. Notebooks may be viewed on the desktop or iconised. The area of the desktop where iconised notebooks appear is known as the 'bookshelf'. Both notebooks and individual pages may also be 'floated' in a window above the desktop.

Input/output jobs are queued by system services. A printout can be initiated even when the printer is not connected. Any number of (named) printers can be simultaneously available. Suitable elements in the printout queue are printed automatically when the appropriate printer device is connected. The system supports auto-detection of connection to peripherals and networks, so that output is processed automatically when the connection is made. Similar facilities apply to disk backup and input of, for example, faxes and electronic mail.

One of the most powerful features of the system is called 'embedding'. This effectively allows views or part-views of one notebook page to be included within another. The mechanism is particularly powerful in that once a given document is displayed through the embedding mechanism, the application responsible for that document is activated whenever it is on display and controls the way the display is managed. Thus it would be possible to embed a dynamically-updated graphical entity (eg a pie-chart updated by network events) within a text document; what would be embedded would not be a particular bit-map (ie a 'frozen' record of the pie-chart data at the time the display was embedded) but rather a dynamic entity which kept itself updated whenever it was on display. Embedding is wholly transparent to the applications programmer.

Handwriting Recognition

Handwriting can be stored either in compressed bit-map form or it can be translated into machine-readable form (the system uses a variety of RTF). At present cursive recognition (reading joined-up handwriting) is yet to be achieved, but hand-print is reliably recognised. GO Corporation has implemented its own recognition engine, which may be trained to individual styles; third-party recognition engines may be substituted. GO has researched this area very thoroughly and claims 95% accuracy. Speed is not a problem, particularly since recognition proceeds concurrently with the I/O handling of pen events.

The handwriting recognition engine (HWX) is typically invoked in one of four modes: dictionary-checked, numeric-only, template-controlled or free.

Our experience suggests that the GO HWX is powerful and fast enough to win wide acceptance. GO itself is however encouraging third parties to compete in this field, in the interests of the quality of the final product. At least one US company is already working on an alternative.

System Architecture

A novel feature of PenPoint is its object-oriented design. About 5% of the code is in assembler; the remainder is written in C, the bulk of which is fully object-oriented code. I will discuss the implementation of the

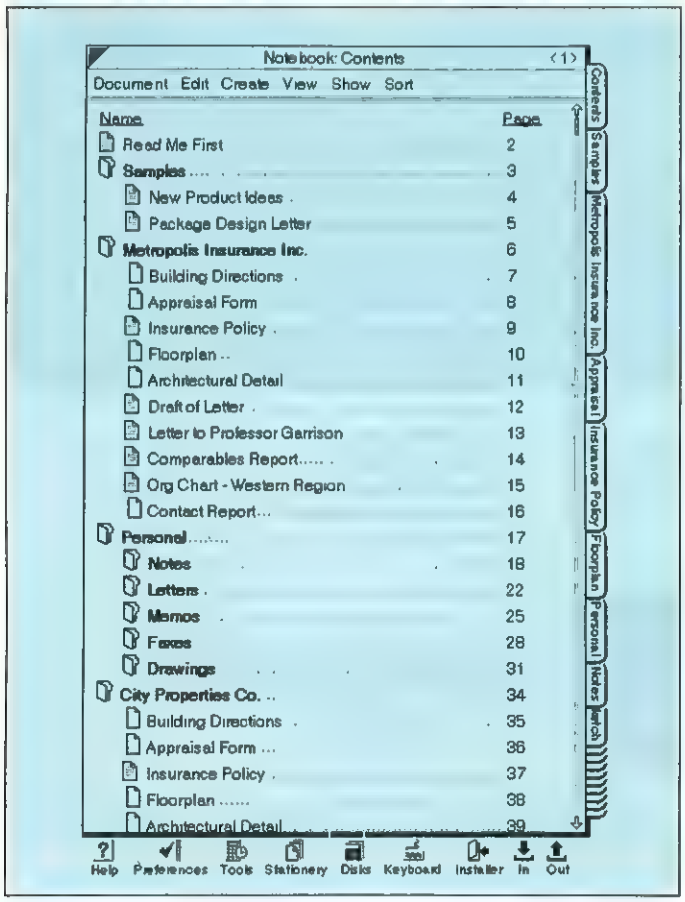


Figure 2 - The PenPoint Notebook

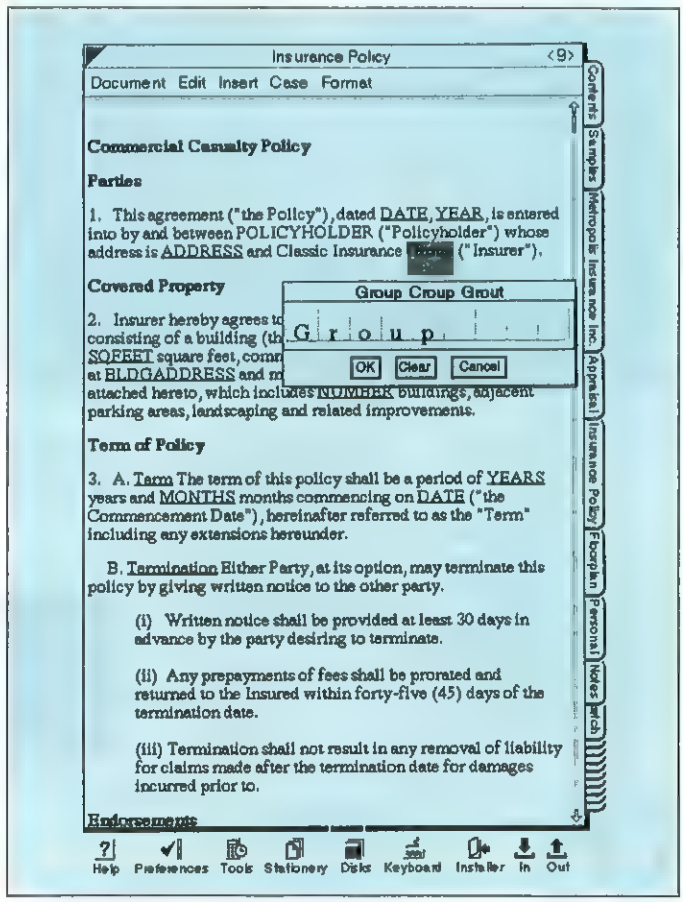


Figure 3 - The PenPoint Edit Pad

BLINKER™ TAKES CLIPPER™ OUT OF THIS WORLD

Blinker eliminates the need for complex overlay structures for both Clipper Summer '87 and 5.0

* Significantly reduces run-time memory requirements
* Accelerates your development cycle 3-10 times faster than any other current linker and even faster when linking incrementaly
* Dynamically overlays C and Assembler code sich as Extend.lib, dGE, FUNCky, other Clipper Add-ons and your own 'C' code
* Optimises your application with advanced profiling functions
* Includes memory defragmentation, 'burning-in' of Clipper environmental variables/serial numbers and creation of demonstration versions

IT'S ALL PLAIN SAILING WITH BLINKER

What the experts say

Here are just some comments from the 5000 happy users to date

"It's blinking fast!"

"Congratulations ... we gained over 100k"

"It's so good, I went through all my programs and re-compiled/linked just for the pleasure of seeing all the free memory"

"It's fast! Handles overlays like a dream"

"Amazing – No more messing about with overlay structures!"

"Brilliant!... But my coffee breaks have been shortened!"

"Actually exceeded our most optimistic performance hopes! (Just for once)"

European sales and support:

QBS Software Limited
10 Barley Mow Passage, Chiswick, London W4 4PH
Telephone 081-994 4842, Fax 081-994 3441, BBS 081-747 1979

Authors and publishers:

Blink Inc Limited
Raglan House, Llantarnam Park,
Cwmbran NP44 3AB

All trademarks recognised

CIRCLE NO 656

"With or without a mouse, it's the cat's whiskers."

New development environments and technology transition make 80's editors inefficient and obsolete. To be competitive, programmers must have leading edge tools. The tool you'll use the most, the one that wraps everything into a single consistent environment will be the powerful, configurable SPE Professional Editor.

Right out of the box, it'll make you more productive, more adaptable, more imaginative. Its instant installation, elegant mouse support, advanced user interface, and point-and-shoot help get you running immediately.

MS-DOS, OS/2 and Dual Mode together

MS-DOS, OS/2 and Dual Mode versions are all included in the same box on both $5\frac{1}{4}$" and $3\frac{1}{2}$" disks, with templates for popular languages, with or without a bundled Microsoft® Mouse. Great value, and your team can move into the 90's with the same editor.

Turnkey emulations

If you prefer the commands and keystrokes of other editors, our turnkey emulations duplicate them precisely. And you still gain the SPE Professional Editor's advanced features, windowing capabilities and powerful engine.

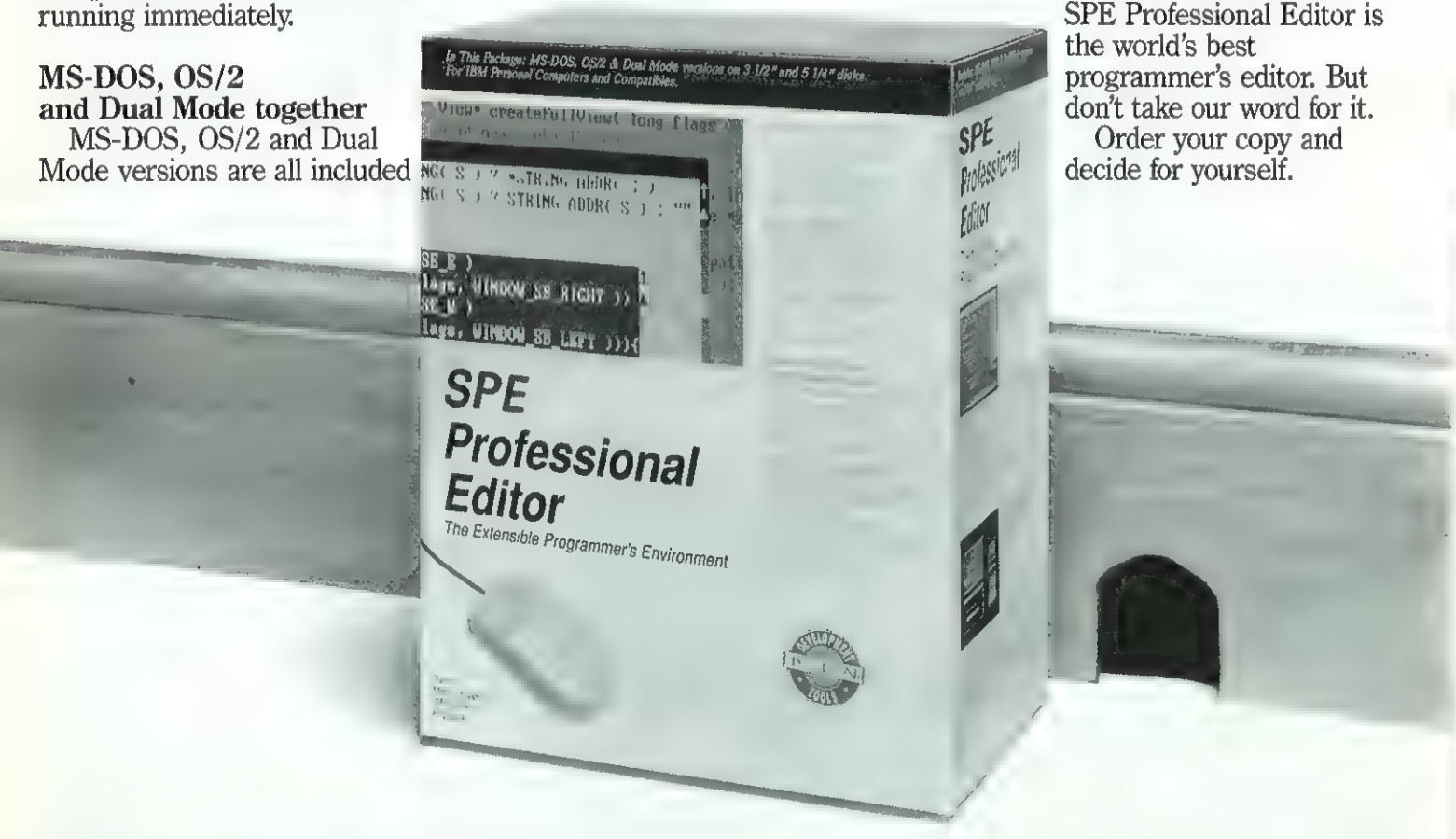
With or without a mouse

Use the editor with or without a mouse — all functions are available without lifting your fingers from the keyboard. But the click-select windows, scroll bars, zoom, shrink-to-icon, block text selects, and other speedy mouse actions make editing extremely efficient.

Don't take our word for it

We're confident that the SPE Professional Editor is the world's best programmer's editor. But don't take our word for it.

Order your copy and decide for yourself.



Works with or without a mouse.
Packaged with or without Microsoft® Mouse.

Version 1.1 now available

APPROVED DEALERS

Absolute Computing Limited, Absolute House
124 Leavesden Road, Watford, Herts. WD2 5EG. Tel: (0923) 242570

Computer Solutions Limited, Canada Road, Byfleet
Surrey KT14 7HQ. Tel: (09323) 52744

Grey Matter Limited, Prigg Meadow, Ashburton
Devon TQ13 7DF. Tel: (0364) 53499

Roundhill Computer Systems Limited, Orchard House
Ogbourne St. George, Marlborough, Wiltshire SN8 1SU. Tel: (0672) 84535

Reflex Technology Limited, 9 Buckingham Place
Bellfield Road, High Wycombe, Bucks HP13 5HW. Tel (0494) 465907

Frontline Distribution Limited, Intec 1, Wade Road
Basingstoke, Hampshire RG24 0NE. Tel: (0256) 463344

System Science, 3-5 Cynthia Street
London N1 9JF. Tel: 071-833 1022

Software Express, Portmill House, Portmill Lane
Hitchin, Herts SG5 1DJ. Tel: (0462) 422525

Software Generation · Portmill House, Portmill Lane, Hitchin, Herts SG5 1DJ

CIRCLE NO. 657

```
Remote Interfaces and File System
  ATP              Provides AppleTalk support
  DirHandle        (eg theWorkingDir, theSelectedVolume)
   AppDir          directory devoted to one application instance
  FileSystem       (eg theFileSystem - used to locate installed volumes)
  Modem            eg fax, data
  ParallelPort     Provides parallel port support
  SoftTalk         Provides TOPS SoftTalk API
  Stream
   FileHandle      Provides file handling support
    ObjectFile     Provides object files for instance data.
    ResFile        Resource files store objects and data by Resource ID.
   Service         general-purpose DLL for network protocols, device drivers,
                   filesystem volume types...
    INBXService    Input mailbox
    OutBoxService  Supports deferred output
     Prn           Implements the abstract functionality of a printer driver.
      BndPrn       Implements a line-oriented printer driver
       Epson       printer driver for a dot-matrix printer
       Pcl         printer driver for the PCL language
       Pscript     PostScript printer driver; can also generate EPSF
                   PostScript files
   Sio             serial port support
   Xfer            inter-task communications
  Transport        remote process communications
  TopsMounter      access to TOPS servers
  Volume
   VolRAM          RAM disk, implements data compression
   VolMSDisk       Implements PenPoint file system on MS-DOS disks
   VolTOPS         Implements PenPoint files on remote volumes via TOPS
  VolSearch        filesystem UI support, eg formatting, duplicating
```

Figure 4 - The PenPoint Object Hierarchy (part)

object system in greater detail, .EXE Editor willing, in a subsequent article.

The object hierarchy is elaborate and of course extensible by the applications programmer. Its object classes fall into the following principal areas:- Remote Interfaces and File System, Text and Handwriting, Applications, Installation and Windows and UI toolkit.

To give the complete class hierarchy is beyond the scope of this article, but details of those classes relating to remote interfaces and file systems are shown in Figure 4.

In principle, much of the work of a GUI application will be done by code permanently present in PenPoint. Because of this level of OS support, a GUI application for PenPoint should be considerably smaller than an equivalent application under Microsoft Windows or OS/2 Presentation Manager.

The file/process system is unusual in that every dataset has an associated process. This is supported primarily by a RAM filesystem, and secondarily by a disk filesystem.

The RAM-resident filesystem employs hardware memory access protection where available. Currently this must be SRAM or non-volatile memory of some kind; GO plans to support DRAM (which is faster and cheaper, but can consume more power) in the near future. The organisation of the RAM volume is roughly mappable on to an MS-DOS or UNIX filesystem, but unlike disk-based filesystems it intrinsically supports data compression. PenPoint document names may include embedded spaces and punctuation, and there are many PenPoint-specific file and directory attributes.

For disk handling, the designers of PenPoint superimposed their filesystem onto that of MS-DOS. The auto-detection facility in the system provides 'hooks' whereby any disk compatible with the drive hardware could be used. Because PenPoint can emulate hierarchical directories, extended attributes and 32-character filenames by means of system disk files, PenPoint can theoretically be hosted by any filesystem. Currently, MS-DOS, RAM and 'remote' (ie networked) volumes are handled; Macintosh HFS is likely to be added.

Graphics I/O devices are supported via a device-independent coördinate system, which apparently owes much to the PostScript language for its base concepts. PostScript-type files are used to specify graphics objects generally. This design offers very 'WYSIWYG' performance. Faxes, for example, are stored internally at their fullest resolution; a fax can be marked-up using the pen at the fullest resolution of the pen digitiser (irrespective of the screen resolution) and retransmitted without any degradation of the image.

Much has been done to make interesting and flexible GUI programming comparatively easy. In particular, the system supports 'smart' window component layout so that, for example, a form can be displayed in either portrait or landscape format without the programmer needing to work out absolute screen locations. The system will arrange multiple elements tidily and sensibly so as to achieve a well-spaced layout under all conditions; for example it will auto-scale or word-wrap boxed text. This minimises reliance on a given graphics video device and improves the portability of GUI applications across different screen and print devices.

The entire system is event-driven. An application's 'main' routine receives control only upon installation and when an application dataset ('document') is created. Subrou-

```
/* method.tbl - contains the method table for clsEmptyApp. */
#include <clsmgr.h>

/* To describe the methods used by the application, we use a
 * method table: a zero-terminated array of struct MSG_INFO.
 *
 * a MSG_INFO contains the message number (a MESSAGE),
 * the handler function name
 * and flags to control the message-passing protocol. */

MSG_INFO clsEmptyAppMethods[] = {
  msgDestroy, "EmptyAppDestroy", objCallAncestorAfter,
  0
};


/* we use a class table to describe the classes used by the application.
 * This is a zero-terminated array of struct CLASS_INFO.

 * a CLASS_INFO contains the name to use for the compiled table,
 * a pointer to the method table (above)
 * and flags (reserved, zero) */

CLASS_INFO classInfo[] = {
  "clsEmptyAppTable", clsEmptyAppMethods, 0,
  0
};
```

Figure 5 - Method Table for Skeleton Application

tines are not called by 'main', but by the operating system upon receipt of some kind of event. All code must be re-entrant and is written to react to specific input events. Typically, an application will define certain responses to certain input events: the 'main' routine will install certain entities in the system and then 'sleep' for ever.

Applications in use on the system are permanently loaded. Because of this, it is not useful to speak of 'loading': rather, one speaks of 'installation'. There is a permanent process for each application which supervises in a global way the interaction of that application with the system. For example, this permanent process can handle installation, de-installation or re-installation, import of datasets from foreign systems, etc. Applications can be informed of specific 'home' directories, which exist on individuated external media (eg a floppy disk with a unique volume ID) onto which they can save/restore their state to free up memory.

The Developer's angle

The PenPoint operating system is remarkable for its user interface and its strong design. Applications written for this system should be smaller and more portable than under any other because of the high degree of user-interface support and device-independence the system offers.

The implementation of an object-oriented OS in C is open to criticism. At the time when GO commenced the project, no suitable implementation of an object-oriented language (and arguably only C++ would have the requisite low-level potential) was available for the target hardware. The object system is closely modelled on Smalltalk ideas. All binding is done at run-time.

The source code for each application consists of two components: a method table, which defines the classes used by the application and the methods responded-to thereby, and the source of the application itself. Figures 5 and 6 show these two source components for a skeletal application. This application creates just one class, namely, the application class. Processes are thought of as multiple instances of an application, so an application is implemented as a class - a subclass of `clsApp`. The class responds to just one message, `msgDestroy`, which is handled by the vacuous function `EmptyAppDestroy`. This is, in effect, the sole method of class `clsEmptyApp`. Before compiling the application source in Figure 6, the method table in Figure 5 has to be pre-compiled by the GO method-table compiler. This produces METHOD.H, which is then included in the C source.

GUI programming is a complex and cumbersome business in C. Programming for PenPoint is certainly no exception, as will be apparent from Figure 6. GO has made extensive use of the preprocessor to simplify the operations of inheritance, specification of function parameters and 'magic numbers' within the system. Unfortunately macros can obscure as much as they clarify; and the C preprocessor is not really up to the task. Already other GUIs have been simplified by object-oriented 'wrappers' supplied by third parties and there is a big opportunity here with PenPoint. There are already some excellent packages appearing (eg PenApps from Slate corporation) which vastly simplify the generation of truly natural pen-based applications.

The Competition

The reason for the obsessive secrecy surrounding the development of PenPoint is now apparent. Days after the launch, Microsoft announced Pen-Windows. A Pen-Windows developers' conference was held recently, with all those attending bound by non-disclosure agreements. What little is known about Pen-Windows suggests that Microsoft is intending to bolt

```
/* emptyapp.c
 * This is a skeletal application: it has no window,
 * nor any state it needs to save.
 * This class does respond to one message
 * (EmptyAppDestroy), so it has a separate method
 * table and a method to handle that message.      */
#include <app.h>     // for application messages
                     // (and clsmgr.h)
#include <appmgr.h> // for AppMgr startup stuff
#include <string.h> // for strcpy().
#include <method.h> // method function prototypes
                     // generated by MT

/* EmptyAppDestroy - responds to msgDestroy */

STATUS LOADDS CDECL FAR
EmptyAppDestroy(
  const MESSAGE msg,
  const OBJECT self,
  const pArgsType pArgs,
  const CONTEXT ctx,
  const pInstData pData
) {
  // this is where any response to the message
  // would be coded...
  return stsOK;
}


/* ClsEmptyAppInit installs the EmptyApp application
 * class and links the new instance to its method
 * table.                                          */

STATUS FAR PASCAL ClsEmptyAppInit(void) {
  APP_MGR_NEW  new;     // struct used to announce
                        // new app manager to the OS

  // Install Empty App class as descendant of clsApp.
  if ((s = ObjectCall(msgNewDefaults,
                      clsAppMgr,  &new)) < stsOK)
    return s;

  new.object.uid = clsEmptyApp;
  new.object.key = (OBJ_KEY)clsEmptyAppTable;
  new.class.pMsg = clsEmptyAppTable;
  new.class.ancestor = clsApp;
  // no instance data: size is zero
  new.class.size = Nil(SIZEOF);
   // no msgNew args of its own
  new.class.newArgsSize = SizeOf(APP_NEW);
  new.appMgr.flags.accessory = true;
  strcpy(new.appMgr.company, "GO Corporation");
  strcpy(new.appMgr.defaultDocName,
                              "Empty App document");
  return ObjectCall(msgNew,  clsAppMgr,  &new);
}

/* main - application entry point (as a PROCESS -
 * the app's MsgProc is where messages show up once
 * an instance is running).
 * This routine gets called in two situations:-
 * at installation, and at instantiation (ie when
 * a new piece of 'stationery' handled by this app is
 * created by the user).  main() can detect which
 * situation by looking at the processCount
 * parameter.                                      */

void FAR CDECL main (int argc, char **argv,
                     U16 processCount)
{
  if (processCount == 0)
  {                     // installation
    ClsEmptyAppInit();   // Create application class
    // get app monitor to install this app
    AppMonitorMain(clsEmptyApp, objNull);
  }
  else
    // create an application instance
    // and dispatch messages
    AppMain();
}
```

Figure 6 - Skeleton Application Source

a pen interface onto its existing Windows product. GO's argument is that a bolted-on pen layer will never allow software developers sufficient system support to produce a truly natural pen-controlled user interface.

Another competitive issue is the consistency of the user interface. PenPoint presents a completely consistent notebook metaphor which requires no understanding of the underlying computer system. Windows has an inconsistent desktop metaphor (when have you seen a trash-can on a real desk? or a Program Manager?). People need to understand computing before they can use Windows; GO says this is not the case with PenPoint.

More significantly, Microsoft is believed to be rewriting Windows as an object-oriented system. When this has been done, Microsoft will certainly be in a better position to compete.

PenPoint is aimed at a new class of machine, portable enough to be usable in a warehouse or an aeroplane, discreet enough for a court-room or a board meeting. GO envisages PenPoint running on machines varying in size from shirt-pocket to drawing-board. The PenPoint desktop machine of the future could have a screen literally the size of a desktop.

Why did GO decide to produce such a radically new system, rather than aiming for compatibility with existing GUI systems? GO would answer: as well as being far easier to control than the keyboard/mouse GUI, the pen-on-screen interface is so radically different, and raises such different possibilities, that the only successful way to produce a good pen-based application is to design it from scratch. Therefore, GO saw the quest for keyboard/mouse applications compatibility as ill-conceived: even if it could be achieved, it would be to the detriment of the pen interface. Rather, the issue is one of data compatibility and device/network connectivity. This issue GO has addressed at a fundamental level of the PenPoint system.

When can I have one?

The PenPoint release demonstrated at the launch was a developers' beta issue. It will run on an 80386 PC-AT for debug purposes, but the ideal development environment consists in networking a GO computer to the AT via TOPS. GO's computer is the size (and about the weight) of a well-built photograph album. It has a 100 dpi LCD screen, but no keyboard! It contains an 80286, typically with around 8 MB of RAM, and runs in Protected Mode. GO has no plans to supply this machine to the mass market.

The commercial V1.0 release, superseding the segmented 80286 version, will run on an 80386 only and will use a flat memory model. The move to the 32-bit world is mainly for portability not speed: GO is keen to have PenPoint running on non-Intel kit. GO is looking to ship V1.0 late in the year.

IBM, NCR and GRiD are all committed to supplying a hardware platform for PenPoint: the race is on to be ready for the commercial release. NCR and GRiD are keeping their options open, and will also be offering Microsoft Pen-Windows when it becomes available.

EXE

CPK Smithies is technical director of Peripheral Vision Limited (Tel. 0373 52755). His language of choice is C++. He is a viola player.

OPEN SYSTEMS PROGRAMMING & PORTABLE SOFTWARE

NEW 32 BIT VLM SYSTEM NOW AVAILABLE

Use Pecan's Power Software development system and gain these substantial benefits.

- *Open systems product*
- *Portability of applications + information*
- *Protection of investment in software*
- *Choice of compiler USCD Pascal, Modula-2, Fortran 77, 'C'*
- *Substantial Cost Savings*

THE ONE COMMON FACTOR

TURBO PASCAL USERS!!
Open the doors to new markets

Move your MS-DOS restricted Turbo Pascal programs to Unix quickly and easily with our Pascal Liberator product.

- *Helps translate Turbo Pascal code to UCSD Pascal code which will then run under UNIX, MS-DOS, CPM, OS/2, APPLE the choice is yours.*
- *Pascal Liberator operates via a user friendly interface, offering pop-up help screens and split screen windowing to display differences between Turbo Pascal and UCSD Pascal command structures.*

Liberate your software for only £230.00+VAT
(This price includes a Power System)

Get ahead of your competition
CALL US NOW

PECAN™

Telephone 0272 - 425012
Pecan Software Europe Ltd. Victoria House, 10 Kellaway Avenue, Bristol BS6 7XR. England.
Fax 0272 - 245000.
or
Distributor Grey Matter Telephone 0364-53499

CIRCLE NO. 658

Taking the XENIX plunge

Richard Samworth was an MS-DOS programmer who needed to get into the big 'U'. Here, with a view to illuminating the path for those who follow, he describes his early battles with XENIX.

In 1980 Microsoft purchased a source code licence for the UNIX operating system from Bell Labs. In those days, it was fashionable to modify UNIX and then release your own version. This is exactly what Microsoft did. The company called it 'XENIX' and it ran on Intel 808X processors. XENIX is now marketed by the Santa Cruz Operation (SCO) as SCO XENIX System V.

Microsoft built some awareness of MS-DOS into XENIX. The operating system comes with a system file called `dos` that loads MS-DOS from the XENIX bootstrap prompt, and a selection of utilities which allow the exchange of files and directory information between the two operating systems. The XENIX Development System has tools for cross compiling applications developed under XENIX so they will load and run under MS-DOS. Later versions of XENIX will also support an OS/2 partition.

Moves towards UNIX standardisation have led SCO to develop the 'merged' product, SCO XENIX System V Release 2.3. Release 2.3 runs applications intended for AT&T UNIX System V/386 without recompiling. Further merging is planned. The 'merged' product lines are only available for 386 processors. Over the years, XENIX has collected a wide variety of PC and PS/2 specific libraries and drivers (described by SCO as 'value-added enhancements'). These are not available with AT&T UNIX System V/386 Release 3.2.

SCO calls XENIX 'the World's most popular UNIX system'. We demur to go this far, but it certainly is one of the most widely-used UNIX systems around.

Hardware

SCO XENIX V will run on Intel 80286 or 80386 processors. In practice, because of the memory management loads imposed by the operating system, 386 processors are preferable.

SCO recommends a minimum of 1 MB RAM for the core system and 500 KB for each additional user. It also suggests that extra memory may be required if the system will be running large applications such as databases or the Development System.

My first mistake with XENIX came with the disk drive specification. I had been told that a 70 MB hard disk was the minimum practical size for anyone wanting to run MS-DOS and XENIX on the same system. For no better reason than that it was large and inexpensive, I decided to buy a Seagate ST-296N 80 MB drive with an ST02 SCSI XT/AT controller board. The disk duly arrived and was installed. The instructions explained that there was a driver to map the device into MS-DOS, so we modified CONFIG.SYS, rebooted the system and hey-presto, acres of fast disk. Then we tried to install XENIX. It wouldn't. Several calls to the disk supplier followed. We changed the DIP switches on the mother board, then the disk controller, then practically everything else. Still no joy. Finally, we got desperate and read the manual. Appendix A, Compatible Hardware, section A.7.8 explains:

Many hard disks will work with XENIX system V Operating System. Whether or not a disk works depends upon the controller board. Here are two tests the controller must meet:

1. The disk controller is fully compatible with the standard controller for that configuration.

2. No special vendor software is needed to make the controller work under DOS.

We are currently using a Seagate 4096 80MB drive with a Western Digital WD1003V MM2 controller.

SCO XENIX V

SCO XENIX System V is a multi-tasking multi-user operating system for IBM compatible micro computers. It comes in a large box and your first task is to assemble the documentation. When you have finished, you should have the *Installation and System Administrator's Guide*, the *User's Guide*, the *Tutorial booklet* and several other shelf-straining binders. You will need to manufacture some space for the manuals (see Figure 1).

I found that you really do have to read the Installation Guide carefully, paying particular attention to Appendix A. This Guide and the Tutorial are the only manuals that don't assume that you know what you are doing. Culture shock intensifies when you reach the Maintenance section of the System Administrator's Guide. XENIX, unlike MS-DOS, needs constant petting and looking after; the System Administrator is the poor innocent charged with system maintenance and operation. In a single-user environment, system administration can be done by whoever happens to be using the system. In a multi-user environment, one individual should be nominated as administrator and made responsible for back-ups etc.

Loading XENIX

In my experience, you should allocate a whole day to the task of getting XENIX up

Name	Width	Weight
XENIX and Development System	1' 6"	1 st 4 oz
AT&T System V documentation	11"	1 st 5 oz
GCI and Text Processing system	6"	4.5 oz
K+R plus two general books	3"	3 oz

Figure 1 - XENIX documentation: the facts

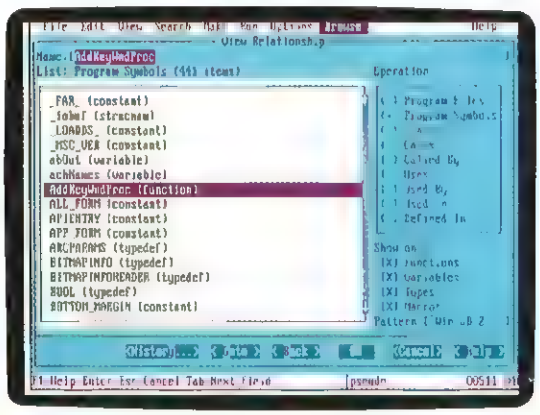
C Breeze.

Get set for smooth sailing. Because Microsoft® C Professional Development System version 6.0 is not only designed to be fast and powerful, but also easy to control.

Thanks in large part to Microsoft's unique Programmer's WorkBench.

It integrates the debugger, editor and compiler, allowing them and all the other powerful programming tools to share data.

Tools such as our Source Browser, that taps directly into the project database. This feature will give you instant access to the relationships between functions, macros, types and variables. It'll even give you a full call tree that literally draws you a map.



Explore the relationships between functions, variables, types, and macros using information created by the compiler and stored in the project database.

With our CodeView® Debugger, you'll save a lot of time. This powerful tool not only debugs any size DOS* or OS/2 application program, on any 286 or 386™ machine, but also lets you explore nested structures and arrays.

The programmer's WorkBench even has an open architecture that is supported by many third party tools.

Our online and hardcopy documentation are designed to complement each other. Together they provide you with an intelligent mix of professional guidance.

You'll find the C Advisor always online and ready to help. It is a hypertext-based reference system complete with sample coding solutions you can copy and paste directly into your program.

You will also find the hardcopy documentation offers many useful tips and techniques for advanced C programming.

So whether you're developing applications for the MS-DOS® Microsoft Windows™ version 3.0 or MS® OS/2 Presentation Manager systems, our Programmer's WorkBench will allow you to write the fastest code around in the shortest amount of time possible.

For a free white paper with more details, just call Microsoft on (0734) 500741.

Then get your hands on Microsoft C version 6.0. It'll blow you away.

Microsoft®

Microsoft Languages and Development Kits may be purchased from your usual dealer or direct from the following Microsoft Language Resellers: Grey Matter, 0364 53499. Software Construction Company, 0763 244114. System Science, 071 833 1022. QA Training, 0285 655888. *This requires 550K of available extended memory. ©1991 Microsoft Corporation All rights reserved. Microsoft, MS, MS-DOS, the Microsoft logo and CodeView are registered trademarks and Windows is a trademark of Microsoft Corporation. 386 is a trademark of Intel Corporation. 386 Max is a trademark of Qualitas, Inc.

and working. Your first act should be to cut all incoming phone lines.

Start by doing a low-level format on your hard disk and partitioning it. If you want to have MS-DOS on the hard disk, it must live in the primary partition. The documentation says, 'starting your DOS partition on cylinder 0 can cause the DOS partition to become inaccessible after installation.' In cylinder 0, MS-DOS writes its boot block next to the end of the XENIX masterboot block. Presumably, under certain circumstances, the XENIX masterboot block can change size.

Having partitioned the disk you can FORMAT, SYS and generally treat the MS-DOS partition as a normal hard disk without affecting other partition(s).

XENIX installation requires much typing in of horrible serial numbers and passwords. These are written on that series of small blue cards that fell under the desk when you opened the bag of disks. Now is a good time to pick them up again and stack them in a neat pile.

Now for the great moment. Put diskette N1 in the floppy drive and reboot. This will load the XENIX bootstrap program:

```
XENIX System V
Boot
:
```

The first decision you have to make is about keyboard type (Norwegian/Spanish/British etc). The boot procedure then loads `badtrk`. If the disk is already partitioned, this should report a DOS partition and one for XENIX. You should make sure that the XENIX partition is active.

If XENIX is being installed onto the hard disk for the first time, you can use the 'thorough scan' option to identify any bad tracks. Don't do this if you are in a hurry: a 150 MB disk will take over two-and-a-half hours to complete. You will need to allocate some space for extra entries in the bad track table, the default is 15.

Swap space allocation can be set to the system default or calculated manually, see Figure 2. Swap space size has a dramatic effect on system performance and can only be changed by re-installing the operating system. It is, therefore, important to get it right.

If you are installing XENIX on a hard disk (or partition) which is larger than 20 MB, you will be asked if you want to divide the XENIX partition into two separate file-systems: root and user. A separate, user file-system (usually known as /u) is not essential, but does offer various advantages. Access to all the files in a file-system is achieved through a structure called the Super Block. The Super Block is maintained in memory and written out to disk every 30 seconds by a process called `update`. There is a UNIX variant of Murphy's law which dictates that the operating system will always crash when the Super Block is most out of date. An unexpected power-off will also damage the file-systems.

During a system crash, a less active file-system will sustain less damage than a more active one. Keeping volatile files in the `/u` file-system reduces activity in the root. Should XENIX crash, it can be re-booted into the root and the /u file-system can be re-created by remaking it and restoring from back-ups made by the system administrator.

Having made the file-system(s), the installation process will ask for the operating system activation key and password.

When these have been entered, the system will reset and may be safely powered off. You can now boot from the hard disk.

The installation continues and copies the Operating System basic utilities from the utilities diskettes. You are prompted for a password for the superuser account 'root'. This password gives its owner maximum privileges.

For time-zone, set 'hours west of Greenwich' to '0' and ignore daylight saving time. The time can be adjusted when necessary from GMT to BST during boot-up. Don't use the abbreviation 'BST', as XENIX takes this to mean Bering Straits Time in Alaska.

If all has gone well, the minimum XENIX run-time system is now installed. If you continue the installation procedure, it will run `custom`, which allows you to install SCO supported products from their respective release diskettes.

Logging In

Each time you juice up the PC, it comes up with:

```
XENIX System V
Boot
:
```

If you have a bootable MS-DOS partition, you can type 'dos' at this stage, and the machine will load MS-DOS without the need to boot from a diskette. Entering 'xenix' (or just pressing the ENTER key at the prompt) starts the XENIX boot procedure.

As we have seen, *stopping* XENIX is more difficult than just switching off the machine. For a start, you must be logged in as 'root'.There are two commands that will legally stop XENIX, `haltsys` and `shutdown`. `haltsys` is abrupt and acts at once; it should be run if you are the only user. `shutdown` is more polite, and closes the system down after a (user specified) number of minutes. It also broadcasts warning 'log off now' messages to all users before killing all the active processes and tidying up. Doubtless many a wag has broadcast a facsimile shutdown message and laughed himself silly at the anguished roars from around the building as people try to save their work.

Systems using major applications (a large relational database, for example) should allocate swap space according to the following formula. The Development System is considered a major application.

1. Multiply the number of users on the machine by the size of the largest process normally run on the machine, in 1 KB. If no unusual processes apply, use 512 KB per user.
2. Take the amount of memory installed and add 256 KB. Compare this result with the result from step 1. Use the larger number as the swap space size.
3. Adjust the number upward if multiple users are running different large applications. Try adding 512 KB per different large application. (Consider the Development System to be a large application.)

For example, the calculations for an 8-user machine with 4MB of memory and a typical mix of spreadsheet, database, word processor and graphics package:

1. 8 users x 512 KB = 4096 KB
2. 4096 KB memory installed + 256 KB = 4352 KB, which is greater than the number calculated in step 1.
3. There are 4 users using different large applications. 4 x 512 KB = 2048 KB. Add this to the greater amount from steps 1 and 2 to obtain a reasonable calculation of swap space: 4352 KB + 2048 KB = 6400 KB. Since the operating system uses a block size of 1K, the swap space allocation translates to 6400 blocks.

Figure 2 - Swap Space Calculation

NEW
Genus GX Support

graphics-MENU 3.0

A sophisticated graphical user interface for Turbo Pascal, Turbo C, Turbo C++ & Microsoft-C.

Automatic use of EMS (expanded memory) if it is available to save the underlying image.

MENUS: several types including horizontal & vertical nested to any depth, supporting keyboard and mouse. Auto save of underlying image.

ICONS: beveled or not, can be used as buttons or as application holders. Can be automatically aligned within a window or isolated.

BUTTONS: automatic arrays allow easy placement in or out of a window. Multiple modes allow use as radio buttons or check boxes.

WINDOWS: optional beveled look. Automatic drag and resizing. Vertical and/or horizontal scroll bars. Title and status lines - easily changed.

FORMS: true data-entry forms available with add-on module. Data validation, range-checking, pop-up menus, help lines, dBASE-like picture strings.

CASE TOOLS: on-screen design tools for complete code generation of menu structures and data-entry forms.

SOURCE: library source is available in all languages.

TASKING: Fore & background tasking capability allows GUI tasks to proceed as a background activity.

Multiple Graphics Environments: Works with BGI (Borland Graphics Interface), Microsoft native graphics, Genus GX graphics or MetaWindow (from Metagraphics Software Inc)

NO RISK: 30-day money back warranty, free technical support, no licensing, no royalties.

£179 graphics-MENU
£379 graphics-MENU with source
£ 99 Data-Entry
£224 Data-Entry with source
£549 ProPak (full source & lib's)
Specify variant: BGI, Microsoft Graphics, Genus GX Graphics or META
Languages: Turbo Pascal 5.5 and 6.0, TurboC 2.0, TurboC++, Microsoft 5.1 (META only), and MicrosoftC 6.0 (META or MS graphics).

Island Systems

Highland Grafix Ltd
Telephone: 0294 69050
Fax: 0294 68286

DEMO AVAILABLE

VISA and MASTERCARD accepted

Highland Grafix Ltd

CIRCLE NO. 660

Compiler Performance

- Based pointers access far data with size and speed of a 16-bit pointer.
- Register-based parameter passing dramatically improves performance of function calls.
- Optimization pragma lets you control what optimizations are in effect from within your code.
- Integrated inline assembler.
- Dramatically improved local code generation.

Programmer's WorkBench

- New Programmer's WorkBench 1.1 is faster than ever before.
- Complete integration of edit, make, debug cycle.
- Source Browser gives you access to information on all aspects of your source code.
- Microsoft C Advisor provides fingertip access to the environment, C Language, and C Runtime Libraries.
- Under OS/2, Programmer's WorkBench takes advantage of OS/2 virtual memory protection, multitasking, and multiple threads to provide you the most productive development environment available.

CodeView version 3.1

- New version 3.1 is compatible with VCPI managers – CodeView can take advantage of extended memory along with VCPI applications like 386-Max.™
- Completely redesigned user interface providing a multi-window, multi-file environment with views into source, data, local and memory.
- Debugs nearly any size application on 286 or 386 machines.
- CodeView takes only 15K from the DOS 640K address space.

Documentation

- Advanced Programming Techniques: Tips and techniques for professionals – to help you get more out of C 6.0.
- C 6.0 Reference: Reference guide covering options and reference to runtime library routines.
- Advisor: Complete reference material online at your fingertips complements the C 6.0 Reference.
- Installing and Using: Gets you up and running with the Professional Development System.

Microsoft®

CIRCLE NO. 661

DAILY NON-STOP DELIVERIES TO EUROPE!

PRE-SALE CONSULTING

AT UNBEATABLE PRICES!

POST SALE SUPPORT

NEW AND NOTEWORTHY:

Adobe Type Manager for Windows ....£ 48
Scaleable WYSIWYG fonts for displays and non-PostScript printers. Supports all typefaces available for PostScript and Display PostScript.

Borland C++ v 2.0 ....£ 270
Windows Development without requiring MS Windows SDK!
Includes Whitewater resource Toolkit.

CLEAR+ for C ....£ 103
Develop and understand your C code better, by automatically drawing program flow charts, multi-level tree charts, formatted source listings, function cross referencing and more!

Gupta SQLWindows v 2.0 ....£ 766
Create Windows and OS/2 graphical applications to access SQL databases the easy way!

Microsoft Windows Productivity Pack..£ 34
Fast & inexpensive way to learn how to use Windows 3.0. Includes tutorial, productivity tips and on-line assistance. Provides fast, flexible and fun learning.

Multiscope Debugger for Windows ....£ 203
Two interfaces: Character mode and as a Windows app. Debug your Windows and DOS applications on a single screen, (no second monitor necessary), on a dual monitor system through a serial cable or even through a LAN network.

Object Vision.... ....£ 75
First ever end-user visual programming system for Windows! Innovative, new visual programming tool that enables non-technical professionals and managers to easily create interactive business applications for Windows 3.0

Smalltalk/V-Windows ....£ 227
Combines a prototype-to-delivery development style while integrating with the host Windows 3.0 environment.

ARTIFICIAL INTELLIGENCE
Arity Combination Pack v5.1 ....£ 594
PDC Prolog (Turbo Prolog) v3.21 ....£ 170
PROLOG-2 Personal v2.15 ....£ 81

ASSEMBLERS & LINKERS
.RTLink/Plus v4.1 ....£ 270
Blinker v1.11 ....£ 130
Microsoft Macro Assembler v5.1 ....£ 79
Plink86+ v2.3 ....£ 216
Spontaneous Assembly v2.0 ....£ 137
Turbo Debugger & Tools v2.0 ....£ 83

BACKUP
Back-It v3.1 ....£ 50
Central Point Backup v6.0 ....£ 49
Copy II PC v5.0 ....£ 22
Copy II PC Deluxe Option Board v5.4 ....£ 94
Norton Backup v1.1 ....£ 80

BASIC
HyperHelp v1.0 ....£ 74
Microsoft BASIC Pro Dvlpmnt System v7.1 ....£ 258
Microsoft QuickBASIC Compiler v4.5 ....£ 54
PowerBASIC v2.1 ....£ 61

BUSINESS
ABC Flowcharter v4.1 ....£ 151
Form Tool Gold v3.0 ....£ 48
Grammatik Windows ....£ 46
Instant OrgChart v1.0 ....£ 101
Lotus 1-2-3 v3.1 ....£ 344
Micrografx Charisma v2.0 ....£ 274
Microsoft Excel-Windows v3.0 ....£ 252
ORG PLUS Advanced v5.0 ....£ 53
PerFORM Pro ....£ 243
Quattro Pro v2.0 ....£ 261
SideKick Plus v1.0 ....£ 107
Super Project Expert v2.0 ....£ 354
TimeLine v4.0 ....£ 371
Xerox FormBase Windows v1.1 ....£ 252

C
Aztec C Cross Development Systems. ....£ 317
Borland C++ v2.0 ....£ 270
C ASYNCH MANAGER v3.0 ....£ 104
C-terp 386 v3.5 ....£ 217
CLEAR+ for C v1.0 ....£ 103
Control Pak/PM ....£ 222
Control Pak/Windows ....£ 222
DESQview API C Library v1.3 ....£ 97
DESQview API Toolkit ....£ 232
Essential Communications v3.0 ....£ 165
Essential Graphics v3.0 ....£ 207
graphicsMENU v3.0 ....£ 115
Greenleaf CommLib v3.0 ....£ 212
Greenleaf DataWindows with Source v2.1 ....£ 270
Guido v1.5 ....£ 130
Intel 386/486 C Code Builder Kit v1.0 ....£ 371
Lattice C Development System v6.0 ....£ 137
Microsoft C Compiler v6.0 ....£ 258
Microsoft QuickC v2.5 ....£ 54
Microsoft QuickC with QuickAsmblr v2.5 ....£ 105
Paradox Engine (C version) v1.0 ....£ 154
PC-Lint v4.0 ....£ 65
Pro-C with Workbench v2.0 ....£ 470
Turbo C TOOLS v6.02 ....£ 83
Vitamin C for Microsoft C v4.0 ....£ 124
WATCOM C /386 Professional v8.0 ....£ 760
Zortech C++ Developer's Edition v2 1 ....£ 245

CADD
3D Drafting ....£ 169
AutoCAD Release 11 ....£ 2390
Design CAD 3D v3.1 ....£ 164
DRAFIX CAD for Windows v1 1 ....£ 346
Generic CADD v5.0 ....£ 191

CASE
Case: PM for C++ v2.1 ....£ 1510
Case: Windows v3.0 ....£ 754
Compeditor II v3 09 ....£ 191
EasyCASE Plus ....£ 215
Teamwork/PCSA v3.1 ....£ 499

COBOL
Microsoft COBOL Compiler v4 0 ....£ 467
RM/COBOL-85 Dvlpmnt System v5.0 ....£ 636

COMMUNICATIONS
Carbon Copy Plus + Host v5.2 ....£ 92
Crosstalk for Windows v1.1 ....£ 101
LapLink III v3.0 ....£ 70
Mirror III v2.0 ....£ 69
Procomm Plus v2.0 ....£ 56
Remote2 (Host & Caller) v2.0 ....£ 90
SmarTerm 240 v3.1 ....£ 158
Takeover ....£ 135

DATABASE & FILE MANAGEMENT
ALPHA/Four v1.1 ....£ 249
AskSam v5.0 ....£ 184
Clarion Professional Developer v2.1 ....£ 383
Clipper v5.0 ....£ 423
dBASE IV v1.1 ....£ 396
dbFast/PLUS v1.4 ....£ 142
dbFast/Windows v1 5 ....£ 203
Documentor v1.5 ....£ 130
FairCom Toolbox Special Edition ....£ 378
Gupta SQLWindows v2.0 ....£ 766
Informix 4GL RDS/IID for DOS ....£ 603
Microsoft SQL Server (1-5 users) v1.1 ....£ 549
Oracle Developers Kit v5 1C ....£ 563
Paradox v3.5 ....£ 412
Paradox SQL Link v1 0 ....£ 283
Q&A v3.0 ....£ 188
R&R Clipper and FoxBASE+ Module ....£ 29
R&R Report Writer for XDB
SQL Media v1.0 ....£ 164
R:BASE 3.1 v3.1 ....£ 396
Superbase 4 Windows v1.2 ....£ 353
UI Programmer v2.1 ....£ 262
XQL Relational Data Manager v2.11 ....£ 446

DEBUGGER
Multiscope Debugger for Windows v1.0 ....£ 203
Periscope II with Breakout Switch v4.0 ....£ 81
Periscope IV-25MHz v4.0 ....£ 1520

DESKTOP PUBLISHING
Adobe Type Manager for Windows ....£ 48
Astral Picture Publisher v2.1 ....£ 274
Facelift for Windows ....£ 51
Freedom of the Press ....£ 203
Go Script Plus v3.0 ....£ 153
HiJaak v2.0 ....£ 110
OmniPage /386 for Windows v2.1 ....£ 353
PageMaker v4 0 ....£ 415
Publish It! v2.0 ....£ 92
Xerox Ventura Publisher Gold v3 0 ....£ 445

DISK/DOS UTILITY
Disk Manager v4.03 ....£ 32
Lotus Magellan v2 0 ....£ 84
Mace Utilities 1990 ....£ 78
Norton Utilities v5.0 ....£ 94
OPTune ....£ 38
PC Tools Deluxe v6.0 ....£ 77
Q-DOS II v2.0 ....£ 31
SpinRite II v1.1 ....£ 53
XTree Pro Gold v2.0 ....£ 77

EDITOR
Brief v3.0 ....£ 150
Brief with dBrief v3.0 ....£ 215
Sage Professional Editor v1.1 ....£ 162
VEDIT PLUS v3.3 ....£ 86

FORTRAN
F77L-EM/32 v3.0 ....£ 566
Fortran Toolkit 1 v5.2 ....£ 60
Fortran Toolkit 2 v5.2 ....£ 99
GRAPMATIC & PLOTMATIC ....£ 132
Microsoft FORTRAN Compiler v5.0 ....£ 235
RM/FORTRAN Dvlpmnt System v2.44 ....£ 302
WATCOM Fortran 77 /386 v8.0 ....£ 643

GRAPHICS
Adobe Illustrator for Windows v1.1 ....£ 229
Autodesk Animator v1.0 ....£ 244
Corel Draw v2.0 ....£ 315
Cricket Graph for Windows ....£ 101
Designer for Windows v3.01 ....£ 384
GRASP with ARTOOLS v3.5 ....£ 94
GSS Graphic Dvlpmnt T/kit for DOS v2.16 ....£ 467
GX Graphics v1.0 ....£ 137
HALO v3.1 ....£ 192
HALO Professional v3 0 ....£ 289
Harvard Graphics v2.3 ....£ 254
MetaWINDOW v3.7b ....£ 188
PCX EFFECTS v1 0 ....£ 69
PCX Programmer's Toolkit v4.0 ....£ 170
Pizazz Plus v1.3 ....£ 76
StatGraphics v5.0 ....£ 504

HARDWARE
80387-33 Math Coprocessor ....£ 485
ASCII Internal 2400 Modem ....£ 59
ASCII Internal 9600bps V.32 with MNP5 ....£ 361
Expanz Disk Expander Card ....£ 108
Practical 2400 Pocket Modem ....£ 91
ProScan PS1000 G-scale Full Page Scanner ....£ 412
QuickTel 2400 Internal Modem ....£ 51
QuickTel 4800 V.22bis/V.42 Internal Modem ....£ 97
QuickTel 9600 Internal SendReceive FaxModem ....£ 146
QuickTel 9600 SendFax Internal Modem ....£ 75
QuickTel 9600 V.32bis/V.42 Internal Modem ....£ 376
SatisFAXtion Classic ....£ 317
ScanMan Plus PC ....£ 121
Worldport 2400 Pocket Modem ....£ 203

INTEGRATED
Framework III v1.1 ....£ 358
Lotus Works ....£ 77
Microsoft PC Works v2 0 ....£ 80

LAN & LAN READY
AboveDISC+ (unlimited users) v4.0 ....£ 164
AboveLAN (unlimited users) v1.0B ....£ 203
Disk Manager-N (Novell) v3 04 ....£ 83
ELS Netware Level I v2.12 ....£ 446
ELS Netware Level II v2.15 ....£ 940
LANtastic 10 MBPS Ethernet Adapter v3.0 ....£ 176
LANtastic 10 MBPS Ethernet Starter Kit v3.0 ....£ 380
Networker Plus v2.2 ....£ 119
Saber Menu for MS Windows 3.0 ....£ 211
Xtree Net ELS Level I & 2 v2.0 ....£ 128

MACINTOSH
Adobe Photoshop ....£ 429
Adobe Type Manager v2 0 ....£ 49
Carbon Copy Mac Single Pack ....£ 94
ColorStudio ....£ 504
FastBack v2.5 ....£ 98
Freedom of the Press v3 0 ....£ 203
HyperDesk v4 2 ....£ 40
LapLink Mac Connectivity Pack v3.0 ....£ 74
MacSleuth ....£ 63
Mathematica for Mac II v1.2 ....£ 575
OmniPage v2.1 ....£ 378
Pagemaker v4.0 ....£ 424
Persuasion v2.0 ....£ 263
Smalltalk/V Mac v2.0 ....£ 92
StatView II ....£ 225
SUM II (Symantec Utilities for Mac) v2.0 ....£ 82
SuperCard v1.5 ....£ 161
Think C v4.0 ....£ 134
Think Pascal v3.0 ....£ 134

MEMORY MANAGEMENT
386 Max v5 1 ....£ 61
AboveDISC+ v4.0 ....£ 50
Headroom v2.02 ....£ 61
QEMM Expanded Memory Mngr/386 v5.1 ....£ 50
QRAM v1.0 ....£ 40
Turbo EMS v6.0 ....£ 43
Vcache v5 06 ....£ 40

MODULA-2
EmsStorage with Full Source v1.0 ....£ 42
Logitech Modula-2 Dvlpmnt System v3.0 ....£ 134
Repertoire v1.6 ....£ 69
TopSpeed Modula-2 DOS Compiler Stn v1.17 ....£ 128

OBJECT ORIENTED PROGRAMMING
Actor for Windows w/Language Extns v3 0 ....£ 598
Borland C++ v2.0 ....£ 270
Object Professional v1.1 ....£ 113
Object Vision v1.0 ....£ 75
Smalltalk/V 286 v1.1 ....£ 92
Zinc Interface Library v1.0 ....£ 136
Zortech C++ Compiler v2.1 ....£ 110
Zortech C++ Developers Edition/386 v 2.1 ....£ 539

OPERATING ENVIRONMENT
386/ix Operating System Release 3.2 ....£ 243
AIX Advanced Interactive Executive v1.1 ....£ 467
Concurrent DOS 386 10 User v 3.0 ....£ 242
DESQview 386 v2.3 ....£ 107
DR DOS v5.0 ....£ 112
GeoWorks Ensemble v1.0 ....£ 90
Microsoft DOS v4 01 ....£ 59
Microsoft Windows v3.0 ....£ 80
Open DeskTop v1.0 ....£ 911
OS/2 Extended Operating System v1.3 ....£ 504
OS/2 Standard Operating System v1.3 ....£ 213
VM/386 Multitasking v1.22 ....£ 113

OS/2 PRODUCTS
Asymetrix ToolBook v1.0 ....£ 232
Display Write 5/2 ....£ 198
MS OS/2 Presentation Mgr T/kit v1.2 ....£ 261
OS/2 Programmers' Toolkit v1.1 ....£ 461
Paradox v2 0 ....£ 389
PDC Prolog v3 2 ....£ 407
R:BASE v2.1 ....£ 494
SideKick for Presentation Manager v1.0 ....£ 107
Smalltalk/V-PM v1 2 ....£ 227
TopSpeed Modula-2 OS/2 Compiler v2.0 ....£ 314
Wide Angle for OS/2 ....£ 54

OTHER
Dan Bricklin's Demo II (Prototyper) v3.0 ....£ 137
Guide 3 for PC (HyperText) v3 0 ....£ 302
HyperPAD (HyperCard) v2.0 ....£ 70
Mace Vaccine (Virus Protection) ....£ 50
Microsoft Windows Entertainment Pack ....£ 23
MKS AWK (Language) v3 1 ....£ 64
MKS Toolkit (UNIX like tools) v3.1 ....£ 157
PolyAWK (Pattern matching) v2 3 ....£ 110
PolyLibrarian (Source documentation) v2 3 ....£ 137
PolyMake v3.3 ....£ 91
PVCS Professional (Version Control) v3.3 ....£ 270
RPG II Development System v3 1 ....£ 938
System Sleuth (Diagnostics) v2.1 ....£ 63

PASCAL
Asynch Plus v5.01 ....£ 104
B-Tree Filer v5 04 ....£ 75
GrafDrive Plus Professional v2.0 ....£ 164
graphicsMENU v3 0 ....£ 115
Microsoft Pascal Compiler v4 0 ....£ 158
Microsoft QuickPascal v1.0 ....£ 54
Object Professional v1.1 ....£ 113
Pascal ASYNCH MANAGER v3 1 ....£ 97
Pascal TOOLS and TOOLS 2 v2.5 ....£ 97
POWER TOOLS PLUS v5.1 ....£ 83
SkipJack v1.01 ....£ 183
Topaz v3.0 ....£ 56
Turbo Analyst v6.0 ....£ 78
Turbo Pascal v6.0 ....£ 81
Turbo Pascal Professional v6.0 ....£ 158
Turbo Plus v5.5 ....£ 119
Turbo Professional v5 09 ....£ 78
View 232 v1.0 ....£ 104

SCIENCE & ENGINEERING
Advanced Math Application Pack ....£ 53
ChiWriter v4.0 ....£ 223
Derive v2.0 ....£ 155
MathCad v2 5 ....£ 256
Mathematica 386 v1.2 ....£ 503
Mathematica for Windows v1.2 ....£ 719
Mathpak 87 ....£ 66
SSP/PC Scientific Subroutine Library v1.0 ....£ 207

SCREEN DESIGN & MANAGEMENT
Fansi-Console v3.01 ....£ 84
POWER SCREEN v1.1 ....£ 83
Vermont Views v2.0 ....£ 270
WideAngle v1.0 ....£ 54

TRANSLATORS
BAS_C v3.2 ....£ 205
dBx/dBport with Source v4.7 ....£ 677
FOR_C ++ for DOS ....£ 731
FOR_C with Binary Runtime v3.1 ....£ 367
For_Struct v1.1 ....£ 448

UNIX/XENIX
Grammatik ....£ 156
Informix 4GL RDS/IID ....£ 1869
Interactive ARCHITECH Multi-User Extension ....£ 488
Interactive Architech Series Workstation Developer ....£ 972
Interactive Motif v1.0 ....£ 99
Interactive Motif Development System ....£ 122
Norton Utilities for System V v3.2 ....£ 180
Oracle UNIX/Xenix 386 1-user v6 0 ....£ 1240
RM/COBOL-85 Dvlpt. System v5.0 ....£ 1046
SCO Microsoft FORTRAN Compiler v3 3 ....£ 390
SCO Microsoft Pascal Compiler 286 v3.3 ....£ 373
SCO Portfolio Suite 386 v6.1 ....£ 852
SCO UNIX System V/386 Development System GT/MC v3.2 ....£ 535
VEDIT PLUS for UNIX/XENIX v3.24 ....£ 156

WINDOWS DEVELOPMENT
Asymetrix ToolBook v1.0 ....£ 232
Btrieve for Windows v5.1 ....£ 302
Halo Windows Toolkit v1.0 ....£ 286
hDC Windows Express v3.0 ....£ 43
KnowledgePro for Windows v1.4 ....£ 496
Microsoft Windows Device Driver Development Kit v3.0 ....£ 260
MS Windows Productivity Pack ....£ 34
MS Windows Software Dvlprs Kit v3.0 ....£ 260
Object Vision v1 0 ....£ 75
Smalltalk/V-Windows v1.0 ....£ 227
WindowsMaker v3.01 ....£ 502

We will match any advertised Price!

- Prices include1st Class Recorded Delivery but are exclusive of V.A.T.
- All major credit cards accepted.
- 30 day invoicing for qualified organizations
- Special discounts for large or corporate buyers.
- Prices subject to change without notice.

UK4/91

USA SOFTWARE

16 CONNAUGHT STREET • LONDON W2 2AG • UK
PHONE: 071-402 0722 • FAX: 071-724 9847

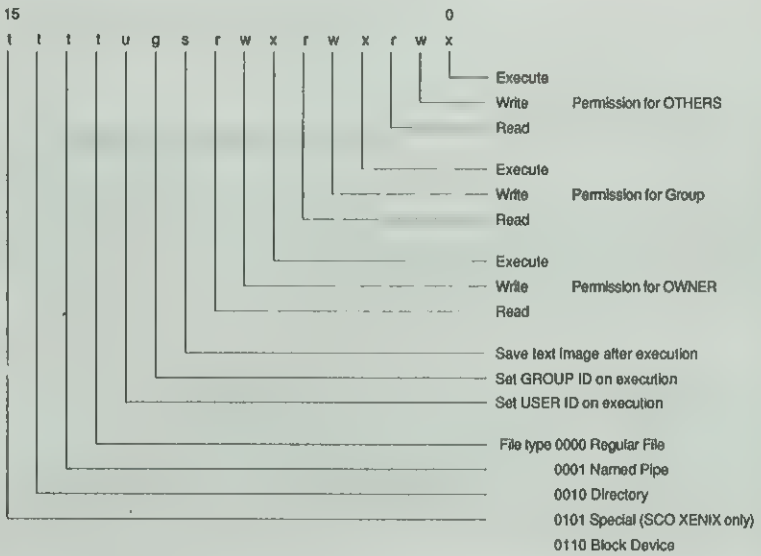
Over 3,500 products carried. Call today for price list-disk.

CIRCLE NO. 662

You can log in to XENIX as 'root' or as a user. The basic system only contains one account: the root. New user accounts are created by logging in as root with the superuser password and using the system administrator's menu-driven shell program, `sysadmsh`, or `mkuser`. When you have selected the appropriate option using `sysadmsh` it will invoke mkuser anyway and prompt for a user id, login name, group, password, shell and a comment. `mkuser` creates a new subdirectory in `/usr`, writes the equivalent of an MS-DOS AUTOEXEC.BAT file (`.profile` for the Bourne shell) and sends the new user some mail. There is a start of day message in `/etc/motd` and you can edit this to contain the witty message of your choice.

The i-node table is an array of 64 byte data structures. There can only be one i-node entry for each file, device and directory in the file-system. From the inode structure (`/usr/include/ino.h`) it is possible to determine everything about a file except the pathname(s) by which it can be referenced.

The i-node structure contains the following entries:



`di_mode`	File type and security keys.
`di_nlink`	Number of links currently referencing this inode.
`di_uid`	The owner's User ID number. This number is defined in the file `/etc/passwd`. It may be changed with the command chown.
`di_gid`	The owner's Group ID number. This number is defined in the file `/etc/groups`. It may be changed by the command chgrp, and is not necessarily the group number that might normally be associated with the User ID number in the previous field.
`di_size`	File size in bytes.
`di_addr`	An array of up to 10 block addresses. Files larger than 10 blocks have additional 'indirect' blocks which, instead of file data, contain a table of up to 256 block addresses. The last element in di_addr points to the first indirect block. The second to last element contains points to the second indirect block and so on. When all the elements of di_addr are used for indirect blocks, the file system doubles and then triples the size of the indirect blocks.
`di_atime`	Date and time the file was last accessed.
`di_mtime`	Date and time the file was last modified.
`di_ctime`	date and time the file status was last changed.

Figure 3 - inode

When you have finished, you type 'exit' or CTRL-D to quit from the root shell. This returns you to the login prompt. Now you login as a user and start looking around.

Exploring

When logged in as a user you are placed in a directory called `/usr/<username>`, where `<username>` is your login name. It contains three entries: `.` , `..` and `.<shellfile>`. These all start with a full stop which means that they are 'hidden' and the shell won't display them. To print the current directory type `pwd`. XENIX is case sensitive: Files called `Fred`, `fred` and `FRED` are all different. By convention, all commands and file names are in lower case. To change directories type `cd <directory>`. So far, so good. Typing `cd` by itself will take you back to your home directory. This can be irritating if you are delving deep into something and inadvertently use `cd` instead of `pwd` to find out where you are. Use `ls -l` to list the contents of directories. You can pipe the output to the `more` utility if the directory contains more than a screen full of entries (`ls -l | more`). You can DOS 'TYPE' the contents of files using `cat`, and browse them with the `vi` editor (type `:q!` to quit from `vi`).

To load a new shell, simply type its name. The bourne shell is `sh`, the visual shell `vsh` (wimps and XENIX only) and the c shell is `csh`. To log off type exit or ctrl-d. When you log off from the last shell XENIX returns you to the login prompt.

The XENIX environment is similar to that provided by the MS-DOS command interpreter. The various shells support pipes and filters which can be used to join processes. Processes are normally interrupted by hitting Ctrl-Break. The Ctrl-Alt-Del interrupt, the DOS three-fingered salute, is disabled.

Shells, like COMMAND.COM under MS-DOS, provide an interface to operating system services. UNIX and XENIX shells are much more powerful than the humble MS-DOS command interpreter and can be used to build complete batch and pseudo-interactive applications. Files containing shell commands (scripts) can be run by typing the appropriate shell and file name. In XENIX, if the first line is a colon and the permissions have been set to execute using `chmod` they will run directly from the appropriate shell prompt. UNIX executable scripts start with #! (hash bang).

When delving around you may be excluded access to a particular file. You can increase your privileges by typing `su` (switch user) and entering the superuser password.

```
#ifdef M_XENIX
#define M_TERMINFO
#endif

#include <curses.h>

main()
{
  int c;
  initscr();              /* Initialise curses */
  keypad(stdscr, TRUE); /* Enable keypad     */
  noecho();
  raw();                  /* Sashimi           */
  attron(A_REVERSE);      /* Set reverse Video */
  printw("Hello World\n");
  refresh();
                          /* Wait for Up Arrow */
  if ((c = getch()) != KEY_UP);
  endwin();               /* Reset             */
}
```

compile as follows :

```
cc <filename> -ltinfo -lx
```

Figure 4 - Using terminfo curses

Search path-names for the shells are stored in the environment space. Use `env` to display the current environment variables. The shell will only search the current directory for filenames if the PATH variables have been set to the current directory in the environment space.

When running with superuser permissions, the current directory is taken out of the search path. This is intended to stop people from inadvertently running user programs with superuser permissions. Under these circumstances, you need type '`./<filename>`' to run a program from the current directory.

The games diskette is pathetic.

File-systems

The XENIX file-system is a familiar looking place. Files are organised in tree structured directories referenced with a forward slash character, `/usr/richard/dotexe`. The basic system comes with some subdirectories already created in the root file-system.

XENIX supports mountable file-systems. To access a diskette, for instance, it should be set up as `/dev/rdsk/fd0` floppy using the `mount` command. Once mounted, the drive appears as the sub-directory `/floppy` in the root file-system. It can be removed using `umount`.

File naming conventions are subtly different from those used in MS-DOS. File names are up to 14 characters long, and may contain any character in the coded character set. There are no file extensions. By convention, executable files are kept in the `bin` subdirectory. When I first tried to run the games, they wouldn't work. I was trying to run text files. All I can say in my defence is that something called 'life' in a subdirectory called 'games' might be expected to run life.

All files have attributes, known as *permissions* in XENIX terminology. The permissions are *read*, *write* and *execute*, and are triplicated for a given file's owner, group and 'the world' ie other users of the system. If a file has its execute bit set, and is not a directory, the shell will load and attempt to run it.

Directories, because they are files, share the same permissions. To limit access to all the files in a subdirectory, you can change the directory's permissions to, say, execute for the owner only. This will exclude all other users from that directory and any subdirectories it may contain. Permissions are modified by `chmod` and permission defaults for file creation are set by `vmask`. See Figure 3.

XENIX implements UNIX file name links. A file called 'article' in `/usr/richard/dotexe` can be linked using the `ln` command to another directory entry such as `/usr/pete/xenix.arc` and two directory entries will be created pointing to the same file. Any I/O done on either link will affect the same file. As in MS-DOS, the directory entries . and .. are not files, but links to the directory and its parent respectively. When its last link is removed, a file is deleted. Thus there is no 'file delete' system call in XENIX; 'unlink' is used instead.

ed and vi

The SCO XENIX documentation deals in detail with two text editors called `ed` and `vi`. `ed`, like EDLIN, is a simple line-oriented editor from the long and noble tradition of simple line-oriented editors. If '`ed`' is short for 'editor', then '`vi`' presumably stands for 'vile'. Actually, '`vi`' is just a *nom de command prompt* for a line editor called '`ex`'. If you type '`ex`', it comes up in editing mode. If you type '`vi`', the program runs up in its visual mode. You can prove that it is the same beast, though, because if you press Esc in editing mode, it reverts to visual mode. Hit Esc in visual mode and it just beeps. There is no obvious way of telling which mode you are in, so `ex`/`vi` beeps a lot.

The red-hot UNIX editor `emacs` is available under XENIX; you'll want to get hold of a copy.

Development System

The SCO Development System consists of a command line C compiler and a modest set of tools. The C compiler that we are running (Version 2.2.1) is not ANSI compliant and does not support function prototyping.

Any programming for an interactive, screen-oriented system in XENIX or UNIX involves getting to grips with the curses library. The curses section in the Programmer's Reference lists the various functions. Figure 4 uses terminfo curses to print the BCPL compiler's historic first words, 'Hello World', in reverse video and waits for the Up-arrow key. It took me weeks to find the section in the manual that dealt with extended attributes such as reverse video (Programmer's Reference, System Services, TERMINFO).

The Development System also includes the usual `make` and symbolic debug utilities. These can be combined using `emacs` to emulate something approaching the integrated development environments available in MS-DOS.

Conclusion

In conclusion, I have two words to say to anyone who is following me down the murky path from MS-DOS to UNIX.

Good luck.

EXE

Richard Samworth is a software engineer at Software Components Limited. He is not available on any bulletin board but can be contacted by telephone on 0296 681 972. Richard was using SCO XENIX System V release 2.3.2 kid 5.52 for i80386 (RRP £675) with the Development System Version 2.2.1 (RRP £550). SCO is on 0923 816344.

INTRODUCING THE ALL ROUND BOX

For more information or upgrade details call Borland Customer Services on 0734 321150, or complete and return the FREEPOST coupon to us today.

BORLAND C++® Everything you need for DOS and WINDOWS programming...

Now there's a vastly superior way to write Windows® applications. Borland C++. The only complete C and C++ programming environment for building DOS and Windows applications. Borland C++ comes with a complete set of tools including a Windows debugger, resource editor and compiler, and WINDOWS.H. So you don't even need the Microsoft® Software Development Kit (SDK).

...for only £299.95 with a FREE Windows Programming Book

Buy Borland C++ from your favourite retail outlet for just £299.95* and we'll send you a FREE copy of Charles Petzold's 'Programming for Windows**'. Worth £27.95 (RRP), this text is widely regarded as a must for Windows developers.

*Plus VAT **Sent on receipt of registration card. While stocks last.

BORLAND
SOFTWARE
KNOW HOW

Borland International (UK) Ltd.
8 Pavilions, Ruscombe Business Park, Twyford,
Berkshire RG10 9NN Telephone 0734 320022

Copyright © 1991 Borland International Inc.

CIRCLE NO. 663

Borland C++ features

■ Windows support including MDI, DLL and DDE ■ ANSI C and AT&T® 2.0 C++ ■ Turbo Drive Compilers and Programmers Platform running in protected mode ■ Pre-compiled headers, increasing re-compilation speed by factors ■ Turbo Debugger for DOS and Windows ■ Whitewater Resource Toolkit ■ Turbo Profiler and Assembler

From the makers of Turbo C++,® Turbo Pascal,® Paradox,® Quattro Pro® and Sidekick.®

Please send me the FREE Borland C++ 2.0 Information Package.

NAME _____

COMPANY _____

ADDRESS _____

_____

POSTCODE _____ TEL. _____

☐ I currently use a Borland language product.

Please return to: Borland C++ 2.0, Borland International (U.K.) Ltd, Freepost, RG1 571, Twyford, Berkshire RG10 8BR.

EXE 4/91

Psst...

CLIPPER* GOES TO UNIX

with

softBASE BRIDGE

but don't tell anyone... it'll be our little secret

* dBASE & Foxbase also supported.

Immediately available for Unix/Xenix shortly on AIX, Ultrix, Sparc, VMS and WINDOWS. Don't get caught with your platforms down!

softBASE BRIDGE is a comprehensive implementation of the dBASE, Clipper and Foxbase programming environments. The only differences being... we are multi platform!

A NAME TO REMEMBER

NDT

New Dimension Technologies Ltd
18 Corporation St West
Walsall, West Midlands WS1 4DU
Tel: 021 311 1470/1490
Fax: 021 311 1296

I would like an invitation to a free seminar near me. Please send details.

NAME .....

ADDRESS .....

.....

.....

TEL No. .....

* All trademarks recognised.

CIRCLE NO. 664

From Brobdingnag to Lilliput: Greetings!

Windows does it; Phar Lap do it; even Borland's TASMX does it. Dan O'Brien examines the emerging DPMI standard.

Protected mode and real mode have a terrible time getting on. Trapped in the same machine, they act like bitter divorcees: the big guy has all the memory, and the tiny, twisted old one hogs all the I/O. The DOS Protected Mode Interface - DPMI - is the latest attempt to get them to talk to one another. It has a lot of support: Intel, Microsoft, Phar Lap, Rational and Borland all have members on the drafting committee. Windows implements it, DOS 5 will at the very least support it, and Phar Lap's new 286 Extender is the first of what promises to be a flood of DPMI DOS extenders. But what is DPMI? And why?

Moving Swiftly on

Well, one thing it's not is 'the DOS extender in Windows'. DPMI is a much lower level protocol, and doesn't support many features which DOS extenders provide. It would, for example, require a large degree of thought to get a heavily DOS-based program running in protected mode simply by using the bare DPMI. On the other hand, if you want to write a DOS extender yourself, or an interrupt driver, or access undocumented system variables in protected mode, DPMI will provide you with the tools you need. And if you are writing a protected mode program from scratch, it might be worth considering using DPMI rather than a DOS extender, because of the finer control it allows over real/protected mode switching. DPMI, for example, will allow intensive I/O procedures to run entirely in real mode, switching modes at the beginning and end of the routine. This contrasts with DOS extenders, which generally perform the switch twice in every DOS call. And even on a 386, that can slow things down.

One ring to rule them all

DPMI's main advantage, however, is that it fills the power vacuum that has otherwise existed in protected mode. Protected mode doesn't have the anarchic structure of real mode, where programs can happily ignore operating system requirements. Protected mode was designed to have a ruler, running at the highest privilege level, allocating memory, fending off alien interrupts, and multi-tasking its children fairly and wisely. The plan was, of course, that this overseer would be OS/2. But, with most machines still running under a real mode OS, protected mode programs have had to install their own masters.

This works fine if only one protected mode program runs at a time. But there's not much room at the top, and memory managers such as QEMM and 386MAX, and extended OSs, such as Windows and DR DOS, take all that room at boot-up time. These programs take all the 386 resources, such as the memory descriptor tables and interrupt handlers, and won't relinquish them. Other, later programs, are run at a lower privilege level - too low to install their own mini-operating systems. This is why, traditionally, you could have a program to load TSRs high, or a DOS extended program, but not both. Both needed control of the machine's protected mode MMU. Quarterdeck and Phar Lap's VCPI protocol moderated

```
/* DPMI Startup code for MSC 6.0 */

#include <stdlib.h>
#include <stdio.h>
#include <dos.h>

#define FALSE 0
#define TRUE  1
typedef int BOOL;

void exitDPMI(unsigned exitCode)
{
      asm
        {
        mov al, exitCode
        mov ah, 04ch
        int 21h
        }
}

void fatal(char *s)
{
puts(s);
exit(EXIT_FAILURE);
};

main()
{
  if (!initDPMI())
     fatal("Initialisation Error");
  puts("Now in protected mode");
     exitDPMI(EXIT_SUCCESS);
}

unsigned int parasNeeded;
unsigned int parasSegment;
unsigned int is32Bits;
void (far *goProtected)();

BOOL initDPMI(void)
{
int notOk;
    asm
      {
      mov ax,cs
      mov oldCS,ax
      mov ax, 1687h
      int 2Fh
      mov notOk, ax
      mov is32Bits, bx
      mov parasNeeded, si
      mov goProtected, di
      mov goProtected+2, es
      }
if (notOk)
  {
  puts("Can't find DPMI") ;
  return (FALSE);
  }
if ( dos_allocmem(parasNeeded,
          &parasSegment) != 0)
  {
puts("Can't allocate memory");
return (FALSE);
  }

  asm
    {
    mov ax, parasSegment
    mov es, ax
    mov ax, is32Bits
    }
(*goProtected)();
return (TRUE);
```

Figure 1 - Getting into protected mode

this condition by allowing a second privileged program to cohabit with whatever took control first. It's an uneasy alliance however; both programs have the same high privilege as one another, and are constantly negotiating between themselves for control of the machine. There is still no one in absolute control, and there is still no room for a genuine protected mode multitasking operating system to take control of both.

DESCRIPTION	ENTRY	EXIT	NOTES
Entry and Exit			
These are not INT 31h calls.			
Get Current Mode	AX=1686h INT 2Fh	AX=0 for protected mode	
Get Protected Mode Entry Point.	AX=1687h INT 2Fh	AX=0 if DPMI BX=1 if 32bit CL=processor (2=286,etc) DX=DPMI version ES:DI=entry point SI=no. of paras	See Figure 1 Use XMS call to enable A20
Terminate	AH=4Ch INT 21h	Returns to launcher	Standard DOS call.
Memory Allocation			
Apart from the selector extensions in DX, these calls require and return the same values as their respective DOS calls - INT 21h, AH=48h, 49h, 4Ah.			
DOS Allocate	AX=0100h BX=para size	AX=segment or error DX=equivalent selector	
DOS Free	AX=0101h DX=selector	AX=error code	
DOS Resize	AX=0102h BX=para size DX=selector	AX=error code	
Extended Memory Allocation			
Get Free Memory Info	AX=0500h ES:(E)DI=buffer		See Figure 3
Allocate memory block	AX=0501h BX:CX=byte size	BX:CX=linear address SI:DI=block handle	No selector
Free memory block	AX=0502h SI:DI=handle		Selectors must be freed separately.
Resize memory block	AX=0503h BX:CX=new size	BX:CX=linear address SI:DI=handle	Watch out - address and handle may change!
Selector Allocation			
Allocate Selector	AX=0000h CX=# of selectors	AX=base selector	
Free Selector	AX=0001h	BX=selector AX=selector	Initial CS,SS and DS can also be freed. Use sparingly!
Convert Segment to Selector	AX=0002h BX=segment		
Get Selector Increment	AX=0003h	AX=Increment	Use with AX=0000 and AX=0100 calls
Selector Description Control			
Only descriptors of allocated selectors can be changed. Access rights and limit values can be obtained from the lar and lsl instructions.			
Get Selector Base Address	AX=0006h BX=selector	CX:DX=address	
Set Selector Base Address	AX=0007h BX=selector CX:DX=address		Use with AX=0002h to save reallocation.
Set Selector Limit	AX=0008h BX=selector CX.DX=limit		Limits > 1 meg must be page aligned.
Set Access Rights	AX=0009h BX=selector (CH):CL=access rights/type		Bits 40-55 (40-47 on 286 systems) of descriptor.
Create Data Segment Alias for Code Segment	AX=000Ah BX=code selector	AX=new data selector	
Get Descriptor Entry	AX=000Bh BX=selector ES:(E)DI=buffer		Buffer is 8 bytes of LDT entry.
Set Descriptor Entry	AX=000Ch BX=selector ES:(E)DI=buffer		
Allocate Specific Selector	AX=000Dh BX=selector		0-10h are definitely unused by DPMI host
Interrupts and Exceptions			
All interrupt code and data should be locked. The DPMI host will automatically switch to a locked protected mode stack during hardware interrupts, INT 1Ch, INT 23h and 24h. Software interrupts do not automatically switch stacks. See Figure 3 for exception stack frames.			
Get Real Mode Vector	AX=0200h BL=interrupt	CX:DX=SEGMENT:offset	
Set Real Mode Vector	AX=0201h BL=interrupt CX:DX=SEGMENT:offset		
Get Exception Handler Vector	AX=0202h BL=exception	CX:(E)DX=handler	Valid exceptions are 00-1Fh
Set Exception Handler Vector	AX=0203h BL=exception CX:(E)DX=handler		Default reflection for 0,1,2,3,4,5+7
Get Protected Int. Vector	AX=0204h BL=interrupt	CX:(E)DX=handler	
Set Protected Int. Vector	AX=0205h BL=interrupt CX:(E)DX=handler		
Protected Mode to Real Mode Communication			
All these calls use the same call structure, given here and in Figure 3. The 'flags' variable in BH should have bit 0 set to one if the interrupt controller and A20 line should be reset. This only occurs on implementations which return to real mode, rather than virtual.			
Simulate Real Mode Int. Call with Far Return Frame Call with Iret Return Frame	AX=0300h AX=0301h AX=0302h BL=interrupt BH=flag CX=Stack words to copy cross buffers ES:(E)DI=call structure	ES.(E)DI=modified call structure	
Carry flag is set on exit to indicate error or refusal for all INT 31h calls.			

Table 1 - DPMI Interface Details

This is where DPMI comes in. DPMI provides all the controls that a user application - including DOS extenders - requires in order to control memory and connect with real mode. But these programs now run at a lower privilege: they can ask for control, but they don't necessarily get it. The Global Descriptor Table - the 386 block which controls all the major system resources, like the call gates that can alter privilege levels, the interrupt tables, and the task descriptors - is now hidden away, and is accessed directly only by the host, the program which provides the DPMI calls. This could just be a DOS memory manager, or a fully-fledged protected mode operating system like UNIX, or an inbetweener, like Windows. Other programs must request from the DPMI API access to memory - and the DPMI host can refuse. The interrupt procedure is similarly controlled by the DPMI host.

The King and his pages

By now, DOS programmers may be throwing up their hands in libertarian disgust, but they may rest easy. The intimidating size of the thing (listed in Table 1) should be enough to reassure hackers, and while the system resources are carefully controlled by the DPMI server, client programs can still possess a great deal of control through the API.

A good example of the DPMI approach is in its support for the 386's paging controls. In previous times, when two programs shared control of the machine at the highest level, neither could successfully operate a virtual paging system. If one took absolute control, this left the other in a serious state of ignorance. Generally, the linear addressing that lies behind both real mode segments and protected mode selector maps directly onto the values that travel on the 80x86's address bus. But with paging, a program cannot ascertain where anything is on the bus, and so programmers wishing to access memory mapped devices are largely stuck. And there are some cases, too, when a pre-emptive paging out, or an operating system attempt to page in a segment could result in disaster. It seems obvious that time-critical interrupt handlers should never be paged out, for example. And for DPMI pagers which still rely on DOS for their file operations (like Windows Enhanced), any data that might be touched from within DOS has to remain in memory at all times, as DOS is not re-entrant. So

U.K. VERSION
NOW SUPPORTS BOTH DOS & OS/2, DUAL MEDIA

POWER AND FLEXIBILITY

BRIEF 3.0 increases your productivity for just £199.00*

*VAT extra, FREE delivery

BRIEF has established itself as the leading programme editor on the market today. Now, with the launch of the latest enhanced version – BRIEF 3.0 – it offers even better productivity.

ENORMOUS UNDO POWER

BRIEF 3.0's enhanced UNDO capability takes the fear out of programming. BRIEF 3.0 allows you to undo any command that affects the cursor or the text (including global changes) up to 300 times, even back to the beginning of your session. It's the safety net you need to program with confidence.

REAL FLEXIBILITY

There's built-in support for 36 languages including Assembler, BASIC, C, COBOL, dBase, FORTRAN and Pascal. And thanks to its complete macro language you can make BRIEF 3.0 fit your own programming style with ease: change key assignment, commands, prompts, colours – even the indenting styles.

BRIEF 3.0 – new benefits

- With our new C-like macro language, CBRIEF, C programmers can write macros on the fly and not have to switch between C and a different macro language
- The new source level debugger works with both the original BRIEF syntax and the new CBRIEF macro language
- Maximum lines per file has been increased to 4 billion
- We've added to and improved template editing and smart indenting for ADA, Cobol, Basic, Fortran, Modula-2, Pascal and C
- We've added editable, multiple keystroke macros that can be saved and restored
- You can even save and restore your window set-ups

Plus the features that made BRIEF famous

- Flexible windowing
- Unlimited number of files
- File size limited only by desk space
- Regular expression search and replace

No other editor puts this much power and flexibility in your hands. So if you're looking for extra productivity purchase your copy of BRIEF 3.0 today.

Solution Systems

SOLUTION SYSTEMS are the UK distributors for BRIEF 3.0. When you buy from us you get the authorised UK version of BRIEF 3.0 dual media (3½" and 5¼") plus bonus macro diskette.

Special phone line for Access/Visa card holders. Save time by phoning 0763 244141 between 9.00 and 6.00 Monday to Friday to place your order.

N.B Official purchase orders from local authorities, PLCs and educational establishments are accepted by post or by Fax.
Fax no: 0763 244025

BRIEF 3.0 The Programmer's Editor

Send to: Solution Systems UK, The Maltings, Green Drift, Royston, Herts SG8 5DB

Please rush me my copy of BRIEF 3.0

☐ I enclose a cheque to the value of £228.85 (£199 + V.A.T.)

Orders despatched same day.

NAME ADDRESS

POSTCODE

Please charge my Access/Visa* account by the sum of £228.85 (£199 + V.A.T.)

Card No: _____ Expiry Date: _____ Signature: _____

* Delete as appropriate

CIRCLE NO 665

DPMI provides calls for locking memory (INT 31h, AX=0600h), and mapping linear addresses onto physical memory (INT 31h AX=0800h) - as have protected mode interfaces in the past. The crucial difference between these interfaces and DPMI can be seen in how it is implemented. First, DPMI's mapping calls are *discretionary* - the DPMI host can refuse access. In more security conscious operating systems, DPMI can thus withhold access to certain areas of memory - accesses to memory mapped ports, for example. Second, the client program can *suggest* paging behaviour to the DPMI server: there is a call to inform the OS that a range of pages can be swapped to disc ahead of time (INT 31h AX=0702h). The new V1.0 of the DPMI standard has even more powerful paging controls than this. So, the DPMI host trusts the client a great deal, and the client has a great deal of control over the host's behaviour; but the host does have the final say.

Speaking terms

So what returns does a DPMI programmer get for all this sacrifice? Well, the most immediate boon is that we can now call real mode routines quickly and easily from protected mode. No less than three routes are provided, all taking roughly the same form. After initialising the DPMI system, and jumping into protected mode, the programmer sets up a structure of the form given in Figure 3, and sets the real mode values as he wishes. Then he chooses which stack frame he wishes to emulate: a simulated interrupt via the real mode vector table (INT 31h AX=0300h), IRET return frame via the CS:IP values stored in the structure (INT 31h AX=0302h), or a far return frame via the same (INT 31h AX=0301h). The DPMI changes modes, calls the routine, stores the new values in the same structure, and returns back to the protected mode routine. Simple. In fact, it can be even simpler, because the DPMI soft and hard interrupts in protected mode are now, by default, transferred (or 'reflected') to the same handlers as would run in real mode. In other words, if you do an INT 21h DOS call in DPMI protected mode, it will *do* a DOS call, in real mode, and return happily with the registers set to the real mode values. It's not entirely as simple as that, of course; there is no segment/selector translation, and, as we'll see, stack operations are a little more complex than usual, but, by and large, a client in protected mode under DPMI has much less work to do than a program which takes full control of the protected mode resources. In Figure 1, `puts("Now in protected mode")` actually works.

DESCRIPTION	ENTRY	EXIT	NOTES
Real Mode to Protected Mode Communication			
Allocate Real Mode Callback	AX=0303h DS:(E)SI=routine ES:(E)DI=call struct	CX:DX=SEGMENT:offset of address to call in real mode.	So, calling CX:DX in real mode will call DS:(E)SI in protected mode
The call back itself has the following conditions:			
	Interrupts disabled DS:(E)SI=sel:offset of real mode SP ES:(E)DI=sel:offset of call structure.	Use IRET with ES:(E)DI=Selector:offset of call structure to restore.	Remember to modify CS:IP!
Paging Control Services			
All addresses are linear.			
Lock Linear Region	AX=0600h BX:CX=start address SI:DI=size in bytes		
Unlock Linear Region	AX=0601h BX:CX=start address SI:DI=size in bytes		
Unlock Real Mode Region	AX=0602h BX:CX=start address SI:DI=size in bytes		Don't unlock memory which DOS may touch!
Relock Real Mode Region	AX=0603h BX:CX=start address SI:DI=size in bytes		
Get Page Size	AX=0604h	BX:CX=size in bytes	Usually 4 KB
Highlight Unused Page	AX=0702h BX:CX=start address SI:DI=size of bytes		Page will be placed at head of page out candidate list if demand load paging implemented.
Discard Page Contents	AX=0703h BX:CX=start address SI:DI=bytes to discard		Contents of region will become undefined
Map Physical To Linear	AX=0800h BX:CX=physical addr to use. SI:DI=size in bytes	BX:CX=linear address	A selector must be allocated to access the block
Virtual Interrupt State **Under many implementations of DPMI, interrupts will always be enabled, to allow the operating system to conduct its own servicing. However, the OS will maintain a 'virtual' interrupt state for DPMI clients, which will control whether hardware interrupts will be redirected into user routines. CLI and STI will only effect this virtual bit. PUSHF, however, will push the real processor flags to stack. These calls allow the virtual interrupt flag to be examined. Two have side effects of changing the virtual interrupt state.**			
Get and Disable Interrupts Get and Enable Interrupts Get State	AX=0900h AX=0901h AX=0902h	AL=0 : disabled AL=1 . enabled	Note that AH is not changed by these calls, so to restore the previous state, just INT 31h
Debug Registers			
Set Watchpoint	AX=0B00h BX:CX=linear address DL=size (1, 2 or 4) DH=type: 0=x,1=w,2=r/w	BX=watchpoint handle	
Clear Watchpoint	AX=0B01h		
Get State of Watchpoint	AX=0B02h BX=watchpoint handle	AX=1 : watch point triggered	
Reset Watchpoint	AX=0B03h BX=watchpoint handle		
Extensions			
Get Version	AX=0400h	AX=version BX=flags bit 0 =1 : 386 bit 1 =1 : real mode (not virtual) bit 2 =1 : virtual memory supported bit 3 =? : reserved CL=processor DH=master PIC base interruptin protected DL=slave PIC base interrupt	
Get API Entry Point	AX=0A00h DS:(E)SI=string	ES:(E)DI=entry point others=undefined	
Carry flag is set on exit to indicate error or refusal for all INT 31h calls.			

Table 1 - DPMI Interface Details (Continued)

The DPMI's use of interrupts is worth a closer look. While the interrupt handling procedure is performed within the DPMI host itself, client programs have (again) three different points in which to redirect the flow of control. Programmers can intercept the handler before it branches to real mode (INT 31h AX=0204/5h), or they can link procedures into the real mode interrupt chain via INT 31h AX=0200h. And if the interrupt was an exception, another vector chain is accessible through INT 31h AX=0202/3h.

Figure 2 explains the contortions which an interrupted program takes under DPMI. The flowchart also shows up some of the assumptions that underlie the DPMI specification. While DPMI is happy to run chunks of its client's code in real mode, it is assumed that most of the application will run in protected mode. So software interrupts whose handlers would traditionally be redirected into the core of a user program - the BIOS timer, Ctrl-C interrupt and critical error handler - are all sent into protected mode, even if they actually occur in real mode. On the other hand, other real mode software interrupts call their standard real mode routines without further redirection. Again, this reflects a DPMI assumption that real mode routines will be used for quick, standard I/O - and the faster, the better. Note that even though real mode software interrupts do not normally get redirected, the DPMI standard does provide a limited number of

Think BIG!

Forget 640K. Forget EMS. Forget the 16-bit squeeze. With LPA 386-PROLOG, you get a genuine 32-bit programming environment, which runs directly under MS-DOS and yet is capable of addressing up to 4G (4096M) of memory. All you need is a 386 or 486, and the talent to Think BIG!

Logic Programming Associates Ltd
Studio 4, Royal Victoria Patriotic Building
Trinity Road, London, SW18 3SX, England
Tel: 081 871 2016 - Fax: 081 874 0449

CIRCLE NO. 666

Affordable CASE Tools

Personal-SELECT at £495

A full function PC-based Analysis and Design Tool supporting the Yourdon, Ward-Mellor, Hatley or HOOD methodologies.

Project-SELECT from £995

Links Personal-SELECT users working in projects and teams in a true multi-user environment.

The SELECT product range is developed and supported in the UK by SELECT Software Tools. It is marketed directly and through a variety of third party distribution channels.

Call SELECT Software Tools today to order your personal copy or to request further information.

SELECT Software tools

SELECT Software Tools Limited
Ullenwood Court, Ullenwood, Cheltenham,
Gloucestershire GL53 9QS

Telephone: 0242 226553 Fax: 0242 251491

CIRCLE NO. 669

Desktop FORTRAN

EXPERTS AGREE WHAT TO BUY – THEY JUST CAN'T AGREE WHY!

"The . . . FTN77 compiler has allowed us to port our main-frame applications to 386 micros with minimum effort. The speed of compilation is incredible!. . . The run time performance is fantastic, full support for 32-bit operations and the 80387 coprocessor means our programs run like lightning!"
Tony Fitzpatrick, Exploration Consultants Ltd.

"I find the FTN77/386 compiler to be fast and easy to use . . . . We often download programs under development from a mainframe to the PC to run through this computer."
Rob McLaren
FTN77/386 User.

"Without the FTN77/386 compiler it would have been totally impractical for us to convert our million-line + power system analysis program (PSS/E) to run on a 386 PC."
Wayne B'Rells
Power Technologies Inc.

"FTN77/386 provides more intrinsic functions than any other: functions to manipulate strings, handle screen and keyboard I/O, read the date and time, parse the command line . . Advanced file manipulation functions allow an FTN77/386 program to work with files in a C-like manner". . . Scott Ladd,
Computer Language, Nov. 1988.

". . . I am very impressed and have decided to shift all our production programs over to FTN77. . "
Dr. G.R. Chaplain
NCL Investments Ltd.

"The ease of use, speed, debugging facilities, dynamic libraries with resulting small .EXE files, and extensive library routines made it a joy to use."
H. Gordon Jensen
FTN77/386 User.

"By far the fastest and most complete FORTRAN compiler that I have ever seen or used in the DOS environment".
Tim Wool
FTN77/386 User.

"The debugger was very easy to use. There is an on-line help facility which made reference to the manual unnecessary."
. . . Mike Gunn and Arul Britto,
EXE Magazine, May 1989.

"For sheer speed and sleekness the (Salford) compiler stands out like a Ferrari among Yugos"
. . . Jack W. Crenshaw, Computer Language, May 1990

FAST, FRIENDLY & FULLY FEATURED – *THE PERFECT MIX – CALL US NOW!*

SALFORD SOFTWARE MARKETING LTD. MAXWELL BUILDING · THE CRESCENT · SALFORD M5 4WT.
TELEPHONE: 061 745 5678 · FAX: 061 745 5666

FTN77 386/486 – extending MS-DOS.

UNIX VERSION AVAILABLE NOW!

CIRCLE NO. 668

addresses, which, if called in real mode, switch to a protected mode procedure - selectable using the INT 31h AX=0303 function. Inserting the values returned from this call into the real mode vector table thus traps all interrupts, real and protected. All this however, is rather against the flow of DPMI, which expects more protected mode to real jumps, and the interface is accordingly awkward. The main niggle is that the real mode registers are frozen into the same call structure as is used for real mode calls from protected mode, and then restored when the protected mode procedure IRETs. This allows their contents to be examined and changed by the protected mode program. Unfortunately, one of the values frozen is the CS:IP, which remains at the point just *before* the protected mode call - thus sending the machine into an infinite loop if it remains unchanged. The act of popping the return value off the real mode stack (as the IRET at the end of a real mode handler would do) has to be performed manually by the protected mode handler code. Also, this structure is stored at a fixed real mode address, thus making re-entrant code a little tough to implement. And finally, real mode callbacks suffer the same problem as all protected mode/ real mode communications - the values in the real mode segment registers need references to 8086 fixed, 64KB segments, while protected mode need selectors.

Which brings us nicely to DOS extenders. They, of course, lie on the other side of the modal fence - they only need to trap software interrupts occurring within protected mode. Typically, a DOS Extender will redirect the INT 21h handler to its own routines, before the real mode handler chain is called. The extender will see every DOS call executed by the application. If the call contains a pointer reference, it will be converted to real segments, either by simple conversion (using another set of DPMI calls) or, if the pointers refer to data above the 1 meg line, by copying it down. If the DOS call refers to a buffer length above 64 KB - for instance when saving data - the DOS extender will break this down into several 64 KB blocks, calling the real mode DOS call a number of times. None of these operations is provided by DPMI, although of course implementing them with it is a lot easier than doing it from scratch.

A knack with stacks

Naturally, all this to-ing and fro-ing between modes presents difficulties, particularly in stack handling. Real mode stacks have to stay under the one megabyte line. Protected mode programs want a protected mode stack. Interrupt handlers, as we mentioned before, need a locked stack and real mode interrupts need a locked real stack. And we need yet another, high privilege stack to allow the operating system to run above all of this. Thankfully, DPMI handles most of the change overs for you, but it is important to know which stack is used when. Generally speaking, a protected mode program will initially use the old real mode stack (with the segment values in SS converted automatically to an equivalent descriptor). Naturally, the protected mode program can relocate its stack where it wishes. Switching to real mode (if the user leaves SS:SP set to zero in the call structure) brings a small (but at least 512 bytes) real stack, which is kept in the DPMI host's own real mode scratchpad area. This is always locked into memory. Callbacks from real mode and protected interrupt code run on a separate, similarly locked, protected mode stack. The old stack will be automatically reinstated as soon as the routine returns.

Getting values to cross from one stack to another is not hard: the real mode SS:SP is easily accessible once it is converted into the descriptor:offset form. This is done automatically in the realmode to protected call back, and stored in DS:(E)SI. Calls to real mode routines from protected mode which require values on the stack are catered for by using CX to set the number of words to transfer from one stack to the other. So, you push the values onto the protected stack, call INT 31h AX=030Xh with CX set correctly, and the values will be re-pushed, in the same order, onto the real mode stack before calling.

DISCOVER
UNIX

INTERACTIVE ARCHITECH SERIES

Workstation Developer
Includes ISC UNIX Operating System 2.21, Software Development System, X11 Development, Looking Glass, VP/ix, TCP/IP, NFS, TEN/PLUS, complete documentation, and more.
1-2 user version $1695
unlimited user version $2549

Application Developer
Includes ISC UNIX Operating System 2.21, Software Development System, VP/ix, Text Processing Workbench, & complete documentation.
1-2 user $1440

Workstation Platform
1-2 user $1100

Network Developer
1-2 user $1525

Network Platform
1-2 user $930

Application Platform
1-2 user $675

ARCHITECH Multi-User Ext. $800

Other Interactive Products
ISC Operating Sys. 2.21, 1-2 user $420
ISC Operating Sys. 2.21, unlimited user $675
ISC Motif Development $419
ISC NFS $419
ISC TCP/IP $419
ISC VP/ix, 1-2 user $419
ISC VP/ix, unlimited user $760
ISC Software Development $675
ISC X11 Development $675
ISC PC Interface $165

CALL FOR CATALOG

INTERACTIVE Technical Support
Technical Support is provided on all Interactive products. Extended Technical Supports Programs available. Please call for details.

OTHER UNIX PRODUCTS
Norton Utilities for UNIX $250
LPI FORTRAN Compiler $839
LPI COBOL Compiler $1269
INFORMIX Products ..... Call
SCO Products ..... Call
Lotus 1-2-3 for UNIX/386
10 user $1295

Programmer's Odyssey specializes in the UNIX Operating Environments, offering a complete line of language and development tools, database management, X Windows and application Software. Call for our FREE Catalogue.

Terms: MasterCard/ACCESS, VISA, American Express, and Wire Transfer.
Hours: 9 AM - Midnight, London Time (4 AM - 7 PM N.Y. Time).
All Prices quoted in U. S. Dollars. Check the current exchange rate for price comparison. Prices exclude VAT and carriage.

PROGRAMMERS
ODYSSEY
120 Hudson St., Northborough, MA 01532 USA

• International (508) 393-5560 • Fax (508) 393-1456 Dealer Inquiries Welcome

CIRCLE NO. 711

If you have to get more out of your next C++ Compiler make sure there's a lot more in it

Choosing your next compiler is no mean feat. Whether you're just putting a toe in the water or about to embark on a major development project with C++, you'll need to know that the compiler you choose gives you the productive edge.

***TopSpeed* C++ packs more in**

TopSpeed C++ packs more in, whether you are developing for DOS, Windows 3 or OS/2. Develop mean-and-lean code with TopSpeed's super-optimizing code generator, Smart-Linking and Project system. Then, add the most extensive C++ class library available and you too can now harness the real power of C++.

***TopSpeed* Putting more into standards**

TopSpeed C++ not only complies with the industry standards offering 100% conformance to AT&T V2.1, but also packs in a range of extensions to take your programming into hyperspace!

***TopSpeed* just the environment for C++**

To make the most out of the C++ and the power of OOP you'll need the development environment that packs the most in. TopSpeed's unique Project system frees you to be creative whilst it manages file dependencies, compilation, assembly, linking and much more.

***TopSpeed* C++ in a family of thoroughbreds**

C++ is just one of a family of four compilers that integrate seamlessly into the TopSpeed Multi-language Development Environment. From $99 per module you can custom-build a development system that takes your programming TopSpeed!

For more information call JPI on the number below.

EUROPE
+44 (0) 234 267500

JPI
Jensen & Partners International

USA
(415) 967 3200

CIRCLE NO 670

Utilities and Programming Tools from Roundhill Computer Systems

.RTLink Plus

Pocket Soft's new "**Virtual Memory Linking**" provides the industry's most advanced memory management technology for Microsoft C, Microsoft Fortran and Clipper applications, with no changes to source code. **.RTLink Plus** also includes a profiler and supports multiple nested overlays, EMS or XMS caching and Codeview. It works with any language which produces standard .OBJ files. The normal price is £295.00 but until June 30th, 1991 Roundhill has a **special offer** price for .RTLink Plus of **£195.00** (plus VAT).

The Periscope Company, Inc.

Periscope/EM is a new product that offers a software-only solution for debugging programs which leave no room in the lower 640K. Qualitas and Periscope have worked together to allow Periscope to use protected extended memory when running in a 386 or 486 system with **386MAX** or **BlueMAX**. Periscope/EM costs only **£155.00** (plus VAT). For 386MAX/BlueMAX add £70.00/£90.00 (version 5.11 is required). Roundhill can also supply the full range of **Periscope hardware debuggers** - please call for details.

PANEL Plus II

Roundhill's **PANEL Plus II** screen manager includes an interactive screen design editor, C and MS-Fortran code generators, and a extensive library of user-interface functions. Support is included for pop-up fields and windows, multiple-line fields, scrolling regions, mouse and scroll-bar operations, pull-down menus and custom validation. There are interfaces to popular PC graphics libraries which allows operation in graphics mode under DOS and OS/2. Key assignments and edit functions are **CUA/SAA** compatible. PANEL Plus II comes with **full library source code**, and versions are available for DOS, OS/2 (including DLL), Unix, Xenix, QNX, and VMS. The library source and your application screens can be ported to any of these operating systems with **no royalties**. PANEL Plus II costs **£285.00** (plus VAT) per development licence. Please call for a free demonstration diskette.

Roundhill also stocks many other development tools: From **Copia International**, we have **AccSys for Paradox** and **AccSys for dBase IV** — easy-to-use, high performance interfaces between the database and your own C-language applications. From **Graphic Software Systems**, we have the **Graphics Development Toolkit** and **Graphical Kernel System** for either DOS or OS/2, as well as **XVT** for source-compatible GUI development over a wide range of operating environments. We also stock **Microsoft** language products. Roundhill is an authorised dealer for **Sage** (USA) and can supply the new **SPE editor** for DOS or OS/2 (including mouse support and unlimited undo/redo), and the **PVCS** source control software, which is now available for DOS, OS/2 several Unix platforms. Please call for a full catalogue and price list.

Roundhill Computer Systems Limited
Orchard House, Ogbourne St. George, Marlborough, Wiltshire SN8 1SU, England
Telephone: **0672 84 535** • Fax: 0672 84 525

CIRCLE NO. 669

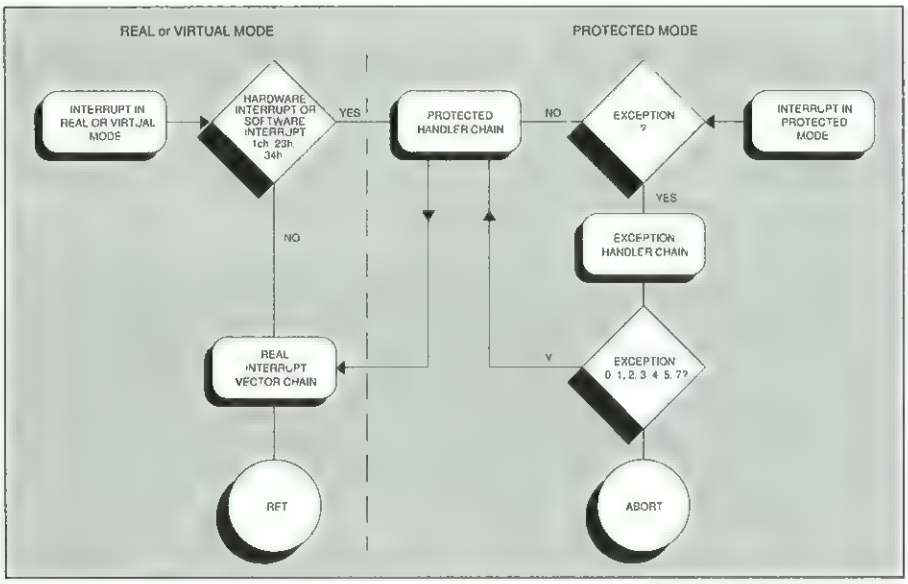


Figure 2 - Interrupt behaviour under DPMI

Malloc aforethought

Whilst communication between modes constitutes the bulk of the DPMI interface, whatever is hosting the DPMI is also controlling the descriptor and paging tables, so it makes sense to provide some simple memory allocation APIs as well. In the interests of keeping the interface low level and flexible,the act of allocating memory is kept separate in the API from actually allocating a descriptor to refer to it. In other words, in order to actually use the memory you have allocated using INT 31h AX=0501h, you must also have allocated a selector for it (with INT 31h AX=0000h) and then set the segment base address (INT 31h AX=0007h) to the linear address you originally obtained. In practice, you'll probably want to allocate a few selectors at the beginning of the program, and simply move their base addresses about, depending on which area you wish to use. A point worth noting is that the memory block itself is resized and freed through a special handle, rather than its address as one would expect.

DOS memory allocation is much simpler, returning both segment and valid descriptor in one call (INT 31h AX=0100h). For reasons of compatibility with 16-bit DPMI implementations, if the block DOS allocated is greater than 64KB in length, DPMI will set up a set of contiguous descriptors, spaced evenly apart. They aren't needed on the 386 - the first descriptor will have a limit equal to the total length so the entire block will be accessible from this selector, but it does aid in using old-time segment arithmetic. The spacing of the selectors can be found using the INT 31h AX=0003h routine. The DPMI specification warns that one 'should not make any assumptions about this value', which going by all previous standards that use this phrase signifies that it will remain eight forever.

That about covers memory management. There are a few rag-tag extra calls, mainly included to keep the DOS extender companies happy. The Allocate Specific Descriptor (INT 31 AX=000Dh) function allows a degree of control over which descriptor value you are allocated. This is so that DPMI compatible extenders can remain compatible with older versions which set aside specific selectors for memory mapped video and so on. Selectors from 0-10h are guaranteed not to be used by the DPMI host. Segment to Descriptor (INT 31h AX=0002h) allows a quickie conversion - but the selectors generated are a limited resource, and cannot be de-allocated. Generally, it is easier to change the base address of a normally allocated selector to the linear address of the area. The code segment alias descriptor call provides for the (very common) situation where the code which initialises the DPMI interface is a small bootstrap program, which then loads the main program into extended memory. To save time, a quick readable and writable duplicate of the new code segment can be made. Note that the DPMI standard itself does not specify a loading format, nor does the API have a load program call. So you'll have to write one yourself. Those interested in how it's done, complete with relocatable code fixups and so on, might like to take a swift dissassembly of the DPMILOAD program supplied with Borland's new TASMX protected mode assembler. Handily, it still has the original symbol data attached.

Watching and waiting

Three more areas of miscellany, and the standard is complete. The first is a generous donation from the DPMI host to the client: control over the 386's built-in debugging features. The debugging registers them-

Offset	Register
00h	EDI
04h	ESI
08h	EBP
0Ch	Reserved by system
10h	EBX
14h	EDX
18h	ECX
1Ch	EAX
20h	Flags
22h	ES
24h	DS
26h	FS
28h	GS
2Ah	IP
2Ch	CS
2Eh	SP
30h	SS

(a) CALL STRUCTURE FOR REAL/PROTECTED MODE COMMUNICATION

OFFSET	16 BIT
10h	SS
0Eh	SP
0Ch	Flags
0Ah	CS
08h	IP
06h	Error Code
04h	Return CS
02h	Return IP

OFFSET	32 BIT
20h	SS
1Ch	ESP
18h	E Flags
14h	CS
10h	EIP
0Ch	Error Code
08h	RET CS
04h	Return EIP

(b) STACK FRAME FOR EXCEPTIONS (INT 31h, AX=0202/3h)

Offset	Description
00h	Largest available free block in bytes
04h	Maximum unlocked page allocation
08h	Maximum locked page allocation
0Ch	Linear address space size in pages
10h	Total number of unlocked pages
14h	Number of free pages
18h	Total number of physical pages
1Ch	Free linear address space in pages
20h	Size of paging file/partition in pages
24h-2Fh	Reserved

(c) FREE MEMORY INFORMATION

Figure 3 - Structures

selves can only be accessed at privilege ring 0, but four calls provided give a useful subset of functions. The calls allow watchpoints of byte, word or double word length to be set (INT 31h AX=0B00h) or cleared (INT 31h AX=0B01). These cause an exception 1 to be generated whenever the selected memory point is written to, read from, or executed (depending on what the watchpoint was set to do). As well as calling the exception handler, the DPMI host will also mark the watchpoint responsible, and this mark can be detected using INT 31h AX=0B02/3h. Watchpoints are not, of course, restricted to debugging purposes.

The raw mode switch utilities (INT 31h AX=0305/6h) allow mode switching under conditions where the already complicated stack arrangements break down. This can occur when a routine will want to call a real mode routine, when it itself is being called by a real mode routine. As mentioned before, the 'other' mode's register state is frozen during a cross-mode call; but only in a static scratchpad store. Calling real mode again would overwrite this state. An example would be when a protected mode interrupt handler called from real mode wants to issue DOS calls. Calling them using the traditional, stack swapping method would overwrite the DPMI hosts own record of the original real mode state. In this situation, INT 31h AX=0305 can be used to obtain pointers to routines that can save these values. The real mode routine (ie that which saves the protected mode's data) is pointed to in BX:CX, and the protected mode equivalent (which saves real mode data) is in SI:(E)DI. AX returns a buffer size. To save the state, call the correct routine with AL=0 and ES:(E)DI pointing to a buffer of that size. To restore, call the same, but with AL=1. Mode switch calls which don't overwrite the DPMI host's own frozen mode values are obtainable from INT 31h AX=0306h. Executing a far *jump* to BX:CX (real mode), or SI:(E)DI (protected mode) will switch modes, but with no registers preserved. There are some continuances, however. The value of AX will be stored in the opposite mode's DS, the old CX will become the new ES, DX will be transferred to SS, (E)BX will end up in (E)SP, and the new modes program counter will start at SI:(E)DI. The (E)BP register will be preserved, and everything else will be trashed. The DPMI specification notes, incidentally, that 'applications may find functions 030Xh more convenient to use than this type of mode switching'. They weren't kidding.

Finally, everything else. INT 31h AX=0A00 takes a string pointer, and returns a pointer to any extended API of this name. This allows DOS extenders to provide extensions to the standard set of DOS calls. Phar Lap has already used this to implement its own extensions.

DPMI is certainly a firm base from which to expand. It will be seen, initially, only for its cross-mode communication features. But, by providing a set of low-level calls that all protected mode programs that wish to use system resources can use, DPMI allows a degree of central control to be imposed on protected mode systems. It also means that there is now a basic set of APIs which should exist under any protected mode operating system - DPMI can (and is designed to) work under UNIX or OS/2 memory management just as easily as it does on its own. It may not be much of a cross-system API yet - but it's a start.

EXE

The DPMI Specification is available (free) by writing to: Intel Literature JP26, 3065 Bowers Avenue, P.O. Box 58065, Santa Clara, CA 95051-8065.

DPMI
OS/2 OS/286 DOS/16M
WINDOWS 3

STRETCH YOUR PASCAL AND FORTRAN PROGRAMS

Running out of memory is a common complaint these days . The first hurdle on most compilers is the 64K limit on code or data. All Prospero compilers have options to leap over this one. The next is the DOS 640K limit. With OS/2 this has never been a problem, and with the new Prospero DOS Extender Kit your DOS programs can now use all the memory on your machine.

The DOS Extender Kit can be used with Windows, DOS/16M and OS/286 and costs £120. It requires Prospero Pascal or Fortran, each costing £240.

Prospero Software
LANGUAGES FOR MICROCOMPUTER PROFESSIONALS
190 CASTELNAU, LONDON SW13 9DH, ENGLAND TEL: 081-741 8531 FAX: 081-748 9344

CIRCLE NO. 670

THE SOFTWARE CONSTRUCTION COMPANY

BORLAND

Borland C++ *NEW*	£199
Turbo C++ 2.0	£55
Turbo Debugger & Tools	£85
Turbo Pascal v.6.0	£69
Pascal Professional	£129
ObjectVision *Until 30.4*	£79
Paradox 3.5	£375
Paradox Engine	£215
Paradox MultiPak	£695

AUTHORISED

COMMS LIBRARIES

C Asynch Manager	£125
C Comms Toolkit	£99
Essential Comms	£215
SilverWare C Asynch	£165
Hold Everything	£129
Asynch Plus	£125
Pascal Asynch Manager	£125

DATA FILE HANDLERS

C-ISAM	£250
c-tree Plus	£385
CBTREE	£129
CodeBase ++	£190
CodeBase ++ Upgrade	£79
CodeBase 4	£190
Essential B-Tree	£129
Paradox Engine	£215
Toolbox Professional	£835
Toolbox Special Edition	£515
Topaz	£65

GENERAL FUNCTIONS

Blackstar C Lib	£65
FUNCky.LIB	£129
Spontaneous Assembly	£129

GRAPHICS LIBRARIES

Essential Graphics	£259
GraphiC	£255
GraphPak	£55
Flipper	£129
GUIDO	£165
Plot 88	£219
dGE	£195
SilverPaint	£85

TEXT SCREENS

C Windows Toolkit	£65
C-Worthy Library Srce.	£175
C-Worthy Upgr to v.2.0	£125
C-Worthy Upgr to v.2.0	£175
C-Worthy/cwArchitect	£389
C-Worthy/cwArc/Srce	£559
Curses by Aspen	£85
Curses by Lattice	£125
Curses with Source	£225
Formaker	£255
Formation Desktop	£129
Formation Dsktp.Srce.	£235
Formation Dsktp Desr	£105
MEWEL Window /Srce	£320
MEWEL Window	£129
Vermont Vws. / Desig	£319
Vermont Vws. GraphEx	£99
Vermont Views / source	£575
Vitamin C	£145

Microsoft

BASIC 7.1	£219
C 6.0 Compiler	£210
COBOL 4.0	£345
FORTRAN 5.0	£199
MASM 5.1	£75
Pascal Compiler	£149
QuickBASIC	£50
QuickC	£50
QuickC/Assembler	£85
QuickPascal	£50
Windows 3.0	£65
Windows S.D.K.	£260

Authorised

CASE TOOLS

EasyCASE Plus	£190
EasyCASE Plus Prof.	£255
EasyCASE Prof. Upgr.	£80
Logic Gem	£65
Visible Analyst	£385
Visible Analyst Wkben.	£2460
Visible Analyst Wkben.	£6995
Visible Analyst Wkben	£1295
Visible Dictionary	£595
Visible Prototyper	£595
Visible Rules	£595

EDITORS

BRIEF for DOS & OS/2	£199
dBRIEF for BRIEF	£99
dBRIEF with BRIEF	£275
Edix	£129
EDT+	£190
EMACS	£210
Epsilon	£129
KEDIT	£99
ME Editor	£59
MKS Vi	£99
Multi Edit	£65
Multi Edit Professional	£115
Norton Editor	£50
PI Editor	£129
Professional Editor	£195
Q Edit Advanced	£39
SPF/PC	£159
VEDIT PLUS	£120

LINKERS

ALINK	£129
.RTLink	£190
.RTLink/Plus	£319
Blinker	£125
Plink	£289
Plink w/LTO	£359

VERSION CONTROL

MKS RCS	£125
MKS RCS -5 User	£485
PVCS	£359
PVCS -5 User	£1345
SMS	£319
Sourcerer's Apprentice	£429
Sourcerer's Apprentice	£175
Sourcerer's Appren.5	£1505
TLIB 5.0	£90
TLIB Network	£195

DEMO BUILDERS

Dan Bricklin's Demo III	£190
Instant Replay III	£99
Instant Replay Prof.	£385

FREE

Microsoft Flight Simulator

WORTH £45 WITH EVERY PRE-PAID ORDER OVER £100*

Proteus	£165
ProtoFinish	£195

MAKE UTILITIES

MKS Make	£99
MKS Make -5 User	£320
PolyMake	£129
PolyMake -5 User	£485

UTILITIES

Above Disk	£79
Batcom	£39
Blue Max	£100
386 MAX Version 5	£85
Copy II PC	£30
Disk Manager	£85
Disk Technician Adv.	£99
DoDot	£99
HEADROOM	£85
HiJaak	£99
Install	£165
Mace Utilities 1990	£95
MultiBoot	£35
Norton Utilities	£95
PC-KWIK PowerPak	£85
QA+ Diagnostics	£129
QEMM-386	£65
QRAM	£55
SpinRite II	£59
Superkey	£55
System Sleuth	£99
The Builder	£99
Toolbook	£249

NETWORK UTILITIES

CC:Mail	£450
CC:Mail Expand	£385
DaVinci eMAIL	£645
LAN Assist Plus	£190
LANshadow	£449
LAN Shell	£320
LAN Select	£320
LAN Space	£320
LAN Spool	£255
Map Assist	£225
Montrix	£515
PS-Print	£255
Saber LAN Adm. Pack	£449
Saber Menu	£255
Saber Meter	£129
SiteLock	£320
X-Tree Net	£255

UNIX TOOLS

MKS AWK	£115
MKS Toolkit	£225
MKS Toolkit-5 user	£640

TERMS & CONDITIONS

PRICING

All prices exclude V.A.T, but include carriage.

DISCOUNTS

Educational & volume discounts available.

PAYMENT

VISA or ACCESS. Credit to approved accounts.

****OFFER***

Applies only to orders exceeding £100 exclusive of V.A.T. on non-discounted products.

TOOLBOX CATALOGUE

Call for free copy.

1 THE MALTINGS, GREEN DRIFT, ROYSTON, HERTS. SG8 5DB
TELEPHONE (0763) 244114 FACSIMILE (0763) 244025

Performance tuning Paradox

Borland's Paradox has built up a good reputation as an alternative to the dBASE-alikes, but it suffers from being available only as an interpreter. Andy Redfern suggests some practical measures to speed things up.

Ansa, and more recently Borland, have taken Paradox from a nice looking idea to an excellent piece of software. They have responded to the changing hardware environments so as to support expanded memory, 386 processors and networks. They have aded VROOMM overlay technology to the package. They have also speeded up the Paradox Application Language (PAL) considerably since its first release, but still one major gripe remains - the lack of a PAL compiler.

Paradox may well provide a fine, interactive, interpreted environment for product development. But at the end of the programming when the system is complete, the developer wants a compiled version which can be sold onto the user. No one likes messing about with run-time versions and run-time licenses. Of course, the other major drawback of using an interpreted version is that the software tends to run much slower than in a compiled version. How many Paradox users have pushed the Do-It! key after typing in a query and waited and waited until you thought the machine had crashed. And just when you were about to despair the answer finally arrives. I have still to find a Paradox user who doesn't want more speed. (Mind you, that probably applies to every other package on the market, but it is especially conspicuous with Paradox.)

So while we wait for the long awaited compiler to be launched, little tips and tricks must be used to get the most out of the current version of Paradox. Of course, it needn't be said that the greatest performance benefits will be found by moving from a 286 machine to a 386 machine. Not only does the software run at a greater clock rate, but the access to the underlying PC hardware is considerably faster. But this is the real world, and often as developers we are forced to use whatever hardware platform the ill-advised or impecunious user has already bought for himself.

External Thoughts

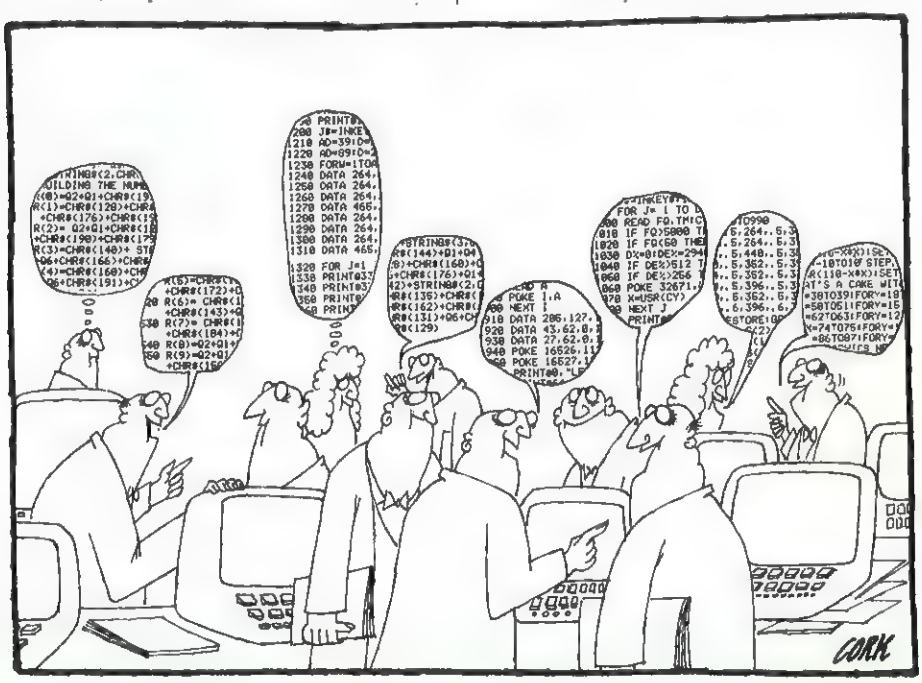
If you're upgrading your system or you are setting up a solution for a customer, there are some important points to bear in mind when choosing the hardware. On machines with 80386 and 80486 processors, the more extended memory you have the better. When installed on any other machine with a 80286 processor or less Paradox works most efficiently with expanded memory. Although Paradox will use LIM 3.2 expanded memory, greater performance can achieved by using either LIM 4.0 or enhanced expanded memory (EEMS). The speed of a DOS `BIG PAL` command can be speeded up an amazing tenfold with just 300 KB of expanded memory.

Paradox is a disk bound application by virtue of the fact that it stores all its overlay, database, script and index files on disk. A disk controller with a hardware cache like the TenTime controller will also provide a dramatic improvement in performance.

Even if you can't justify or afford to buy extra hardware Paradox can be speeded up simply by modifying the way in which you use your existing hardware.

The golden rule is this - Paradox is a database not a disk cache or an expanded memory manager. Paradox may well have the ability to cache the disk itself, but that doesn't escape the fact that it is fundamentally a database program. Therefore, if disk caching is an important performance consideration, buy a specialist disk cache program and let Paradox get on with being a database. This might sound a little harsh on the programmer who wrote Paradox's disk cache, but a few minutes' experimentation will prove the point.

It also worth pointing out that the Paradox disk cache only works on drives C and D.



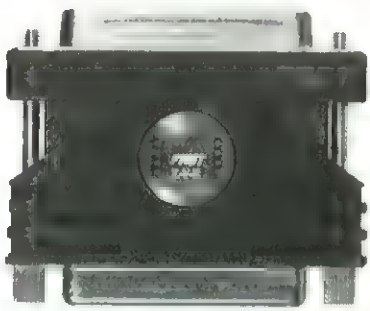
5 out of 5 hackers prefer other software protection methods to Hardlock E-Y-E®

What hackers dislike...
Hardlock E-Y-E was designed using cryptographic principles. It took the experience and know-how of Germany's No. 1 in software protection and the leading edge technology of a US semiconductor company to create the ultimate software protection tool. Hardlock E-Y-E is based on a custom chip featuring secure algorithmic response rather than simple bit swapping or counting schemes.

What software developers like...
Hardlock E-Y-E combines all the features software developers require in a single product: algorithmic response to provide security and an optional non-volatile memory to allow custom configurations. FAST Electronic has made implementation of Hardlock E-Y-E in your software easy. Use HL-Crypt to protect .EXE or .COM files, or incorporate high level language interface routines in your software. The algorithm parameters and the contents of the memory can be programmed in seconds using our Crypto-Programmer card. This unique card guarantees that no one else can burn your original codes. Simply plug the card into any PC slot and start up your own Hardlock E-Y-E workshop.

What your customers will like...
Hardlock E-Y-E allows unlimited backup copies. The device is shipped with the software for the user simply to plug into the parallel interface and forget. Daisy chainability, outstanding reliability (no battery is needed), and the most compact High-Tech design ensure that your customer will accept Hardlock E-Y-E.

What your accountant will like...
Hardlock E-Y-E needs no factory coding. This ensures optimum delivery schedules and stock flexibility. Revenues will go up as software piracy and multiple usage are prevented. Despite its wealth of features, Hardlock E-Y-E's prices remain competitive.



Hardlock E-Y-E programmable, algorithmic response and memory option – all in one.

As more and more software developers, customers and accountants appreciate the Hardlock E-Y-E device, hackers like it less and less.

MAGNIFEYE

Order your demo unit today. Contact Magnifeye,
235-239 Walmer Road, Walmer Studio #6, W11 4EY, Telephone 071 221 8024, Fax 071 792 3449.

CIRCLE NO. 672

It doesn't work on network drives, nor does it work on drives with more than 32 MB capacity. It therefore makes sense to use a cache program like PC-Tools' PC-Cache or PC-Kwik. Then in the custom setup program (type PARADOX3 CUSTOM to load it) set the percentage amount of EMS used for disk caching to zero.

In the Paradox documentation you will find that it requires 512 KB of RAM to run, but as with everything else, the more RAM you can give it in the conventional area the better it runs. On a machine with no expanded memory, the difference between 509 KB free at the DOS prompt and 609 KB free was considerable. Paradox ran around three times faster on a sort or query with that extra 100 KB of memory, and this is a major bottleneck for many users.

Paradox can be run stand alone, but many people use it on a network. In this configuration, the file server carries all the files while the local machines are kept to a minimal hardware configuration to make the multi-user system cheap to install. Once the network drivers have been installed, there is often hardly enough memory left to run Paradox as well. The answer is to use a memory manager like Quarterdeck's QEMM or 386MAX. Unfortunately, these two products only work on 80386 or 80486 machines, so you'll need a product like Quarterdeck's QRAM to get most out of the memory on a 80286 or 8086 machine.

These products all work in a similar way, allowing the user to load drivers, system files, DOS resources like FILES and BUFFERS and TSR programs into the high memory between 640 KB and 1024 KB. Exactly how the drivers are loaded high and into which areas of memory they can be squeezed will depend on exactly what type of hardware you have. Having said that, most machines have enough room to load the Novell or Token Ring drivers high.

Paradox setup

Once you have maximised the DOS memory and disk usage, we can begin looking at exactly how we set up Paradox itself.

The speed of a DOS BIG PAL command can be speeded up tenfold with just 300 KB of expanded memory

As I mentioned before, Paradox is a very disk intensive program so I recommend monitoring the program's disk access very carefully. Purely through experimentation I have found that once my hard disk falls below 96% unfragmented, performance degradation follows. This is because Paradox is having to work much harder to piece together all the files it has created. I now run Norton's Speed Disk (part of the famous Norton Utilities) on a regular basis and find the performance difference quite marked.

Furthermore, we need to monitor very carefully exactly how many files there are in each directory. In the original versions of DOS there was a limit of 128 files in any directory. To allow more files to be added to a directory supplementary file lists can be added. These have a disk access overhead associated with them. This problem can be partly improved by using the DOS FASTOPEN command. This command keeps track of the last few files that were accessed, so that if an application is repeatedly using the same files, the access time is significantly improved. The memory overhead of FASTOPEN is around 200 bytes per file. It can be loaded into high memory or in DOS 4.0 it can use the /X parameter to load itself into extended memory.

Although FASTOPEN will provide a performance improvement, the simplest way of reducing the file access overhead is to keep the number of files in any directory used by Paradox below 90. This way even when Paradox creates its own temporary files and tables during sorts and queries, the total number of files will not exceed 128. It may sound simple, but it certainly works!

Keeping the number of files in any directory below 90 is only possible providing that you use subdirectories to separate out the files, but here lies another trap for the unwary. Both DOS and Paradox can very quickly access files in directories near the top of the directory structure, but once you get more than four levels down the performance of file access once again degrades. To this end, all UNIX hackers should avoid creating directories more than four levels deep, unless you use very short directory names.

Inside Paradox

To enhance the performance of data access within Paradox, the use of indexes or keys is crucial. When you create a key field, Paradox not only spots non-unique entries, but also keeps the table of keys sorted constantly. This is a time consuming business. To make it easier for Paradox bear the following points in mind.

Keep the size of the unique keys to an absolute minimum. For example, if you need to key on a person's name, key on the surname plus a unique number, rather than the surname plus the first name and/or title. The smaller the unique key, the faster the system will run on both searches and updates. Better still, just key on the unique number. This will speed up data entry and if you need to search on the surname on a regular basis, why not add a secondary index. Hard disk space is relatively cheap after all.

The secondary indexes are created using the Query Speed Up option. This takes whatever query is currently in the work space and creates a secondary index that

Products mentioned

QEMM and QRAM are available from Quarterdeck (0245 496699).

386MAX is distributed in UK by International Data Security (071 436 2244) priced £79.

PC-CACHE is included with all versions of PC-TOOLS. PC-TOOLS Deluxe V6.0 costs £115 - details from publishers.

Central Point Software International are on 081 848 1414

PC-KWIK is bundled free on some machines (Dell for example) or is available in its latest version (Super PC-KWIK Power Pack) from Corporate Software (0734 845361) priced £72.44.

Paradox 3.5 has a retail price of £595. Borland is offering an upgrade deal to users of dBASE, DataEase, R:BASE and FoxBase of £149.95.

The Paradox Engine is available directly from Borland on 0734 320100.

C++ DEVELOPER'S EDITION V2.1

For total portability right now, your choices are a bit limited.

Only Zortech Gives You Total Portability to MS-DOS, Windows, OS/2, DOS 386, UNIX 386. And Macintosh. Right Now.

What platform will your application be running on this time next year?

Now that C++ has made significant inroads into mainstream application development, you must choose the C++ compiler that is going to be the performance and portability leader now...and years to come. Since 1986, we've been testing, refining and perfecting our family of C++ products tirelessly. The results are what you are reading about, a fully portable family of C and C++ products available right now.

We Intend To Keep Up This Pace.

In the past few years, Zortech has quietly become an industry standard earning acclaim from Byte, Computer Language, Dr. Dobbs, and many industry experts for our solid implementation, market-leading performance, and strong, relentless innovation.

Whatever Platform You Choose, Choose Zortech Performance.

With Zortech, no matter what platform you choose to develop on, a substantial performance increase will be realized due to the quality of the original Zortech C++ implementation. For instance, on MS-DOS, your program will compile up to 35% faster with the resultant code running up to 45% faster and 25% smaller than Turbo C++*. But we don't stop there. On each platform, Zortech C++ is designed to improve your productivity where it really matters... in the development cycle.

For example, using the DOS/MS Windows package, you can edit, compile, link and debug the largest MS windows applications...and never once leave the Windows environment. All this plus complete plug-and-go to CNS C++/View and Commonview II! And, of course, these advantages are also available on the OS/2, DOS 386, UNIX, and Macintosh platforms.

60 DAY MONEY-BACK GUARANTEE

Choices, Choices, Choices...

If there is one constant in C++ development, it's change. The Zortech family of products keeps your options exactly how they should be. Open.

To find out more, call today.

Order Hotline:
44-81-316-7777

Zortech

Industry Standard. Industrial Strength.

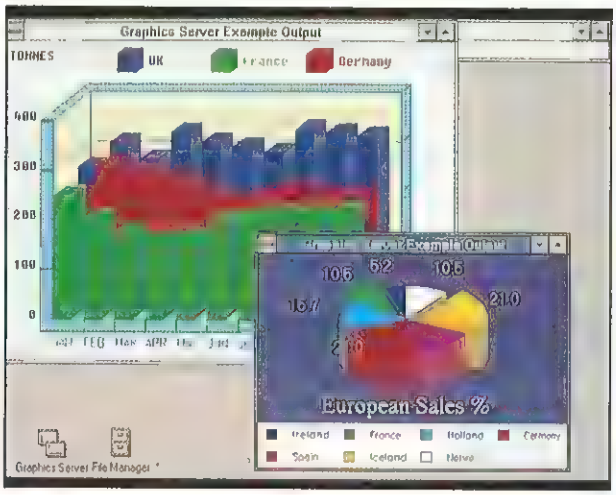
Zortech, Inc., 4-C Gill Street, Woburn, MA 01801 Tel: (617) 937-0696 Fax: (617) 937-0793
Zortech, Ltd., 58-60 Berkford Street, London SE18 6BG Tel: +44-81-316-7777 Fax: +44-81-316-4138
Supported systems include Compaq, ISA/EISA, IBM PS/2 and 100% compatibles

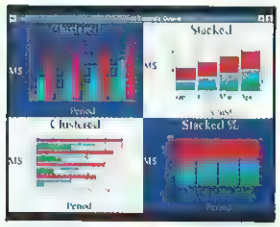
*Source: MS-DOS, C++ Compilers, A comparison by the Ladd Group, (c) 1990

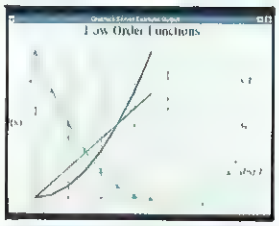
CIRCLE NO. 673

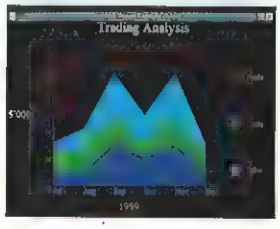
Bring Graphs and Charts to your Windows Applications Licence-Free with Graphics Server™

Introducing the **Graphics Server Software Development Kit (SDK)** from Bits Per Second. The most cost-effective, time-efficient way to integrate full-colour graphs and charts into your Windows applications. Here's why.









Client-Server Model

Graphics Server is a Windows server process which responds to multiple, concurrent requests for interactive graphing and charting functions from client applications. It enables graphs and charts to be incorporated into any Windows application having a Dynamic Link Library (DLL) or Dynamic Data Exchange (DDE) capability. This includes C programs, Superbase, SQL Windows, ToolBook, Word for Windows and many others. In fact, virtually all Windows applications!

Multiple Choice

Graphics Server lets you choose from 16 graph types, including 2-D and 3-D bar (simple, stacked and clustered), 2-D and 3-D pie, area, Gantt, scatter, polar, log-log/log-linear, line, high-low-close, X-Y and polar, plus a variety of statistical functions.

Ease of Programming

Graphics Server, through its multi-level interface, provides the level of control **you** want. In as few as **eight** calls you can open a new graph window, define titles, legends and labels, transfer data, draw a chart, and output the chart to the clipboard or printer. Graphics Server takes care of all the details; scaling the graph to match the data, matching fonts, text size and position to the graph, even resizing and repainting the graph window when necessary! You'll save development time. And money.

Alternatively, if you want full control yourself, Graphics Server provides over a **hundred** functions to help you design, create and manage virtually any kind of graphically drawn image. Multiple graphs, multiple windows, user interaction via hot regions. All these and more!

Full Support

The Graphics Server SDK gives you all the tools, documentation and examples you'll need to make immediate and effective use of Graphics Server. Within hours your applications will be transformed by the inclusion of presentation-quality graphics.

Telephone/fax support and a bulletin board service are provided for all registered users, and for power users requiring an even higher level of service and security there's a special VIP Support Programme.

No Licences!

The Graphics Server SDK costs £275 + VAT. No further licence fees to pay.

To place an order, request a demo disk, or obtain additional information, call . . .

BITS PER SECOND

Bits Per Second Ltd
14 Regent Hill
Brighton BN1 3ED
Tel (0273) 727119
Fax (0273) 731925

All product names are trademarks of their respective holders.

CIRCLE NO 674

PVCS Professional Version Control System

If these are your questions:

1. What is the difference between today's code and yesterday's?
2. ZYX have version 2.1.1 of ABC, where is the source code?
3. How can Jack access (read-only) Jill's source code?
4. What is the latest version of module PQR?

We have the answers:

1. **PVCS**
2. **PVCS**
3. **PVCS (Lan Version)**
4. **PVCS**

By archiving successive versions of source code, in a project archive file with information on the author, date and time, **PVCS** tracks all revisions in a project.

LAN and multi-user (coming for 386 Unix) versions permit access to be controlled on a module by module basis, preventing two programmers from modifying the same file.

PVCS functions integrally with **PolyMAKE** and **PolyLIBRARIAN** and **SPE Professional Editor**.

Special Offer

Purchase (before 31 May 1991) a **PVCS** network license (5,10, 15 users etc) and get a **free** copy of **SPE Professional Editor**, the editor that retrieves from and restores to **PVCS** archives. Of course, the SPE Professional Editor

- Is fully featured for programmers
- Has C, Clipper, Pascal language support.
- has unlimited UNDO, REDO
- can edit multiple files
- has multiple windows,
- has full mouse support

Contact your specialist for PVCS and SPE Professional Editor

System Science 3-5 Cynthia St, London N1-9JF **(071) 833-1022**

CIRCLE NO. 675

will speed up that particular query. In a recent test with a 3 MB database, Query Speed Up managed to cut a query which took 1 minute 20 seconds down to just 12 seconds.

As your application is going through the design phase, it is worth bearing in mind that Paradox is written in such a way that databases which are 'high and narrow' rather than 'short and fat' can be searched much quicker. Put another away, keep the number of fields within a database as low as possible and don't worry about how many records you may then need to create. From the point of view of data access, it is much better to have two databases that are 80 characters wide connected by a unique key than to have a single 160 character wide database. The recommended maximum number of characters wide a database should be varies from one developer to the next, some recommend figures as low as 60 characters while others claim that there is no problem until the database exceeds 127 characters. I prefer to keep them less than 80 characters wide, but this has more to do with keeping all the data on the screen than it has to do with any proven performance cut off point.

Network Speed

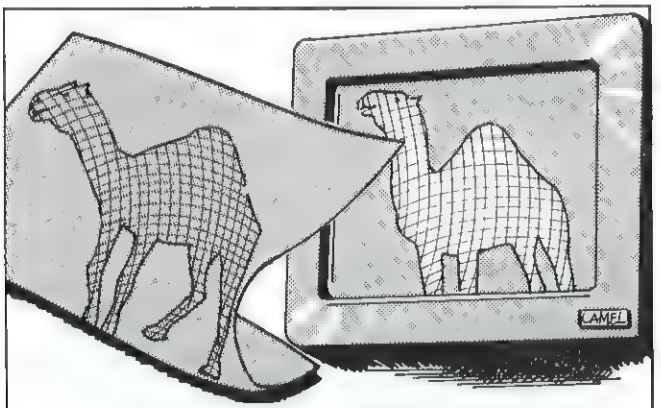
One of the major reasons that networks are so attractive to the corporate buyer is that the cost of installing 10 stand-alone PCs is considerably greater than the cost of installing a single file server and hanging 10 lower specification PC workstations off it. The problem for the Paradox developer is that remote drives, network traffic and lower specification machines all slow the application down.

Keep variable names as short as is practical, and try to spread them evenly throughout the whole alphabet

The biggest component of the network traffic is Paradox's use of overlay files. Wherever possible locate these in local storage. The overlays, help and help index take up around 1 MB, so it is feasible to use a RAM disk instead of having a local hard disk. To make sure Paradox always find its overlay files, list their location as the second item on the DOS PATH. The first item should be the location of the Paradox directory itself.

You can also locate any data files that are only used by the local user on the local machine. You could also make a copy of any data files that are read by the application but are not written to. This does have the disadvantage that a single data backup no longer saves all the data used by the application.

Another tip is to buy two machines of the same specification as the file server. This allows you always to have a reserve server in case of emergency and to have a 'power station' on the network. Then in any application you develop use the power station for any complex operations that need to be performed. Use it for complex queries, on-line back-ups and connect the network printer to it, so it runs the network print spooler as well. This



PLOT THE DIFFERENCE

PLOTVIEW

Is the simple, low cost alternative to paper plots. Able to preview plot images on a graphics screen within seconds.

PLOTVIEW allows paperless plotting and replotting – avoiding delays, improving efficiency.

PLOTVIEW

Has up to 25 separate plot files previewable simultaneously on screen and a zoom function which allows any part of the plot to be inspected in detail.

Easy to integrate, PLOTVIEW is ideal for system builders and can be customised to suit their individual needs.

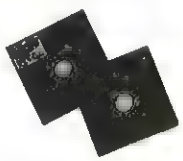
Phone now for more details about PLOTVIEW: the versatile tool that can really make a difference to your productivity.

CAMEL SERVICES LTD.
Telephone OXFORD (0865) 512678

CIRCLE NO 676

BRENT
Communication and Telemetry Consultants

"MaxPro – the ultimate in hardware for software protection"



The MaxPro Data key fitted to the Parallel Port is totally transparent to the end user.

Software customers make as many backups as are required but of course only the machine using the individually coded MaxPro key will run the program.

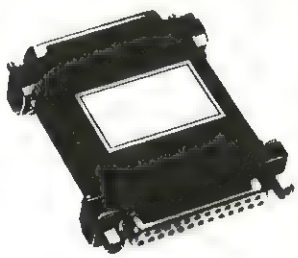
BEWARE THE PIRATE'S PATCH
High security encryption is created by the Simple-to-Use Menu-driven MaxPro attachment software. Other data keys may be patched out in the software rending it totally vulnerable. MAXPRO protected files cannot be patched. Choose whether or not to shut down the program if tampering occurs.

Optional limited life and Usage information collection facilities are provided.

No source or object code modules needed.

The MaxPro system works on IBM; PC; XT; AT Compatibles and PS2.

For additional information contact us at:



Brent Communications
Unit 2
Dragon Industrial Estate
Harrogate HG1 5DN

Tel: (0423) 566972
Fax: (0423) 501442

CIRCLE NO. 677

power station helps to liberate some of the tedious tasks from the local machines and keeps the file server free to serve files!

Faster PAL

All these optimisations and performance considerations provide an excellent foundation for speeding up Paradox, but exactly how you use PAL can be the difference between a slow and fast application.

The first tip is inherent within the Paradox architecture - PAL is interpreted so all the variables and labels have to be searched and looked up on the fly. Paradox uses a simple hashing algorithm to store and retrieve the variables from memory. This means that keeping the variable names as short as is practical and trying to spread them evenly throughout the whole alphabet makes a noticeable improvement.

On a memory limited machine, almost as soon as a PAL procedure has run it is switched out of memory. Also the available resource memory can often be reduced to 0 KB, therefore making it very cumbersome to open a new image on the screen. Paradox allows you to manipulate the amount of resource memory that is left through the `SETSWAP` command. Some of the Paradox text books recommend setting this to somewhere in the region of 40% of memory left (using the `MEMLEFT()` procedure). I would rather advise that you keep it to an absolute minimum at all times. The smaller it is, the better chance Paradox has of not having to switch procedures out to disk.

How you use PAL can be the difference between a slow and fast application

Another useful swapping tip is that if a procedure is continually being switched out of memory, take it out of the library and locate it in the application itself using the `PROC` command. Then it can only be switched out of memory when you actually `RELEASE` it. Beware though, many an 'out of memory' has been generated by trying to hold onto too many of the machine's resources.

Wherever possible, try and keep the number of procedures in any library down to around 50, and aim to make every procedure as small as possible. For Paradox to make the most efficient use of its swapping, try to keep them under 5 KB in length and for best results aim for 3 KB.

If you're working on a single table, Paradox provides two mechanisms for searching for a relevant record. You can Query By Example (QBE) or you can simply `SCAN` through all the records. Surprisingly enough, in some circumstances a `SCAN` can be quicker than a QBE and it is always worth entering a piece of test code and comparing the two techniques. This seems to be especially true on tables with many, short records.

If you're using the `IF/ELSE/ENDIF` or the `SWITCH/CASE/ENDSWITCH` constructs with the `SCAN/ENDSCAN` con-

"AS YOU LIKE IT!"

Copy Protect AND/OR
Copy Register...Don't give it away & cry!

EVERLOCK:
Stops copying & controls programs as you want it.

EVERTRACK:
Does not copy protect but it does control.

EVERLOCK EVERTRAK

E V E R L O C K

- COPY PROTECTS YOUR PRODUCT
- POSITIVELY SECURE AGAINST 'BIT-COPY' SOFTWARE
- PREVENTS DE-BUG, TRACING & DISASSEMBLY
- SUPPORTS ALL HARD & FLOPPY DISC FORMATS
- COMPATIBLE WITH MOST NETWORKS
- OPTIONAL RESTRICTION OF NETWORK USAGE
- LIMITS PROGRAM USE BY EITHER NUMBER OF DAYS AND / OR EXECUTIONS
- REMOTE RESET OF INSTALL, DATE, EXECUTIONS
- SUPPLY UP TO 99 INSTALLS ON ONE DISC
- NO DAMAGED MEDIA OR NON STANDARD 'DOS' ACCESS
- SUBSTITUTE I.O PLUG FOR SOFTWARE
- END USER TRANSPARENT
- EASY PRODUCT UPGRADE
- "SECURE STRING"- EG USER REGISTRATION / SERIALISATION

EVERLOCK NEW RELEASE POWER

E V E R T R A K

- NON - COPY PROTECTION OF SOFTWARE
- AUTOMATIC PRODUCT SERIALISATION / REGISTRATION
- ANTI - REVERSE ENGINEERING
- OPTION TO ATTACH PASSWORD
- LIMIT NETWORK CONCURRENCY
- DYNAMIC PROGRAM COMPRESSION
- DYNAMIC PROGRAM ENCRYPTION
- END USER TRANSPARENT
- COMBINE EVERLOCK & EVERTRAK

EVERTRAK NEW RELEASE POWER

DUPLICATION SERVICE

ANY QUANTITIES LARGE OR SMALL

WITH OR WITHOUT PROTECTION

HIGH QUALITY STATE OF THE ART EQUIPMENT

TURNKEY PRODUCTION AVAILABLE

AS YOU WANT IT!

USER FRIENDLY

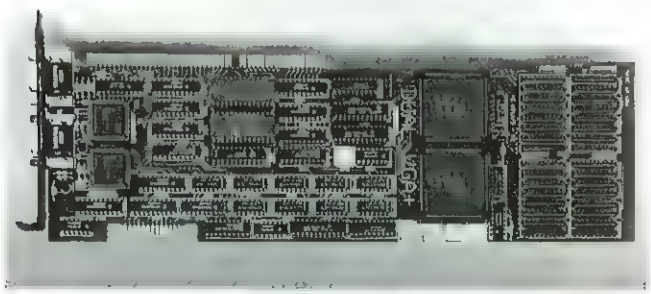
Tel: 0527 585 550
Fax: 0527 65111

ICS House, Edward Street, Redditch, Worcestershire, B97 6HR England

CIRCLE NO. 678

MULTIPLE SCREEN CAPABILITIES USING A SINGLE PC . . .

THE DUAL VGA PLUS



★ UP TO 800×600 RESOLUTION
★ TWO INDEPENDENT SIMULTANEOUS DISPLAYS WITH OVERLAY CAPABILITY.
★ 100% COMPATIBLE WITH IBM VGA/EGA/CGA/MDA VIDEO STANDARDS
★ 4 BOARDS IN ONE SYSTEM GIVE 8 SIMULTANEOUS DISPLAYS

NOW INCLUDES WINDOWS 3.0 DRIVER AND MULTIPLE TERMINAL EMULATION PACKAGE

CEBRA COMMUNICATIONS LIMITED
26 LORNE PARK ROAD, BOURNEMOUTH
TEL 0202 299048 FAX 0202 299192

CIRCLE NO. 679

The definitive data protection software for micros and micro-networks

MAIN FEATURES

- All data held on a hard or floppy disk or a network can be kept secure from unauthorised access.
- Security is by default and is failsafe. Once CLAM has been set up the user does not have to take any positive action to secure data.
- Protected files are kept in a permanently encrypted form on the disk. Even if there is a power or hardware failure the data will never be readable by an unauthorised user.
- A log of all activities can be kept.

Microft Technology Limited
The Old Powerhouse, Kew Gardens Station
Kew, Surrey TW9 3PS.
Telephone: 081-948 8255

£148 + VAT Single User.
Site and Corporate Licences available.

CIRCLE NO. 680

struct, make use of the LOOP command where you can. If there are mutually exclusive conditions don't just let the procedure drop through to the ENDSCAN statement and then LOOP - force the LOOP as soon as you can. The LOOP command simply jumps back to the first line of the SCAN. You can also, where possible, use the QUITLOOP command to jump out of the SCAN once all the records you want to find have been found. Use this technique carefully though - too many condition checks looking for the QUITLOOP condition will again slow the procedure down, therefore negating any benefit of the QUITLOOP itself.

If you have experience of programming in C, then the Paradox Engine provides a neat way of speeding up time consuming routines. The Paradox Engine allows Paradox tables to be manipulated by C programs directly. Using the RUN BIG command these can then be incorporated into your PAL programs. Version 2 of the Paradox Engine, which should be available in the next few weeks, will also support Turbo Pascal and Windows programs, thus really opening up the possibilities.

Finally, it is worth reminding yourself that you are not dealing with a clever C or Pascal compiler that can spot all the standard optimisation techniques. If you want a routine to be optimised for speed you are going to have to do it yourself. No automatic shifting of invariant code out of loops here. If you're not sure of how to optimise a routine for speed, there are plenty of books on the subject of writing optimal code.

In the real world, developers are forced to use whatever hardware platform the user has bought

I really feel like I've only just scratched the surface of what you can do to make your Paradox applications faster. My final suggestion is that you get a stop watch and experiment. Many of the things I have mentioned can be improved upon simply by altering the order in which an operation is done. Queries for example, can often be optimised simply by splitting them down into simple steps.

Do also bear in mind that many of things I have suggested may not actually make any difference to the application you're writing. For example, if you have EEMS memory, the actual amount of conventional memory you have is not quite as significant as if you've only got LIM 3.2. Many operations are not actually slowed down by reducing conventional memory if you have LIM 4.0, but in all other circumstances the difference will be more than worth the effort. So get a stop watch and start experimenting.

EXE

Andy Redfern is a freelance journalist who has contributed to DESQview User, Computer Intelligence/Europe and Microscope.

DESkey
THE COMPLETE DONGLE RANGE

Data Encryption Systems Limited have more experience in the design of software protection devices (dongles) than any other company in the U.K. *12 years* experience in the design of dongles have gone into our latest 2 products: The DK1 and The DK12, both of which use the very latest in ASIC technology.

- 128 BYTES of memory split into 2 x 64. *Half is intended for both read and write, the other half is read-only. Writing to this protected half can only be performed using your unique password. This feature can be used for tasks such as identifying the modules of a multi-module package which have been paid for.*
- Down Counter. *This enables a customer to be sold 'n' goes of the software. After which the DK1 stops working until the counter is reset using your password.*
- Pseudo Random Number Generator. *Billions and billions of bits without repeating! Software and Data Encryption could not be easier. 'Seedable' too.*
- Completely Transparent to Printers, End Users and anything else requiring the port. *16 or more on any one parallel port.*
- No Hidden Extras. *Everything you need is in the DK1. No need for extra 'programming' units.*
- DESlock Software *uses DK1 or DK12 to provide instant encryption of .EXE or .COM software without the need for access to source (or .OBJ). Will even protect DOS commands. Keeps you/your employees the right side of the copyright laws.*
- Designed and manufactured by the U.K.'s leading software security specialists *Data Encryption Systems Limited.*
- DK12 *is a reduced feature DK1, available at a quantity price to beat ALL the competition (ask for a quote).*
- Serial and Bus versions available.

For further information, contact us at,

Data Encryption Systems Ltd

Edbrook House, Cannington, Bridgwater, Somerset. TA5 2QE
Telephone (0278) 653456 Fax (0278) 653300

CIRCLE NO 681

Objective dBASE programmers try C++

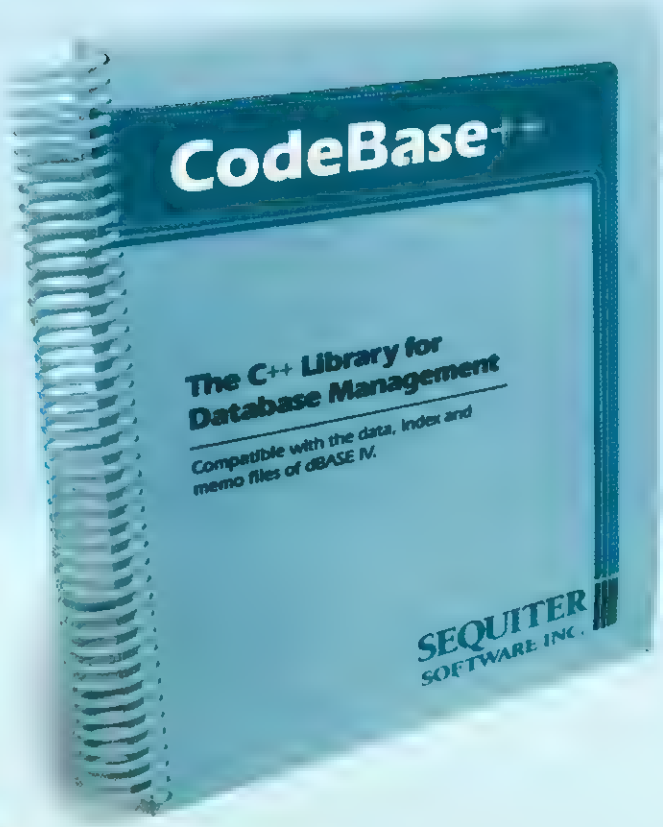
Use CodeBase++ with C++ to build multi-user, dBASE compatible programs. Take advantage of object orientated programming to quickly create maintainable, readable software.

dBASE Compatible

Use CodeBase++ classes to access and change the data, index and memo files of dBASE IV and III. Use other classes to manipulate strings, sort at high speed, perform date arithmetic, evaluate dBASE expressions, manipulate linked lists and manage memory.

Portable

Port your programs between DOS, Microsoft Windows, and OS/2. Keep all the profits as you distribute your programs royalty free.



Fast

Watch as your C++ programs execute much faster than corresponding dBASE, Clipper or FoxPro programs. Be surprised at the small executables and low memory requirements. Examine the C++ source code, which is completely included, to learn good object orientated programming techniques.

Order Today

Order today at £195 by contacting your nearest software dealer. Discover why Sequiter Software Inc. and participating dealers offer a 60-day money back guarantee.

SEQUITER SOFTWARE INC.

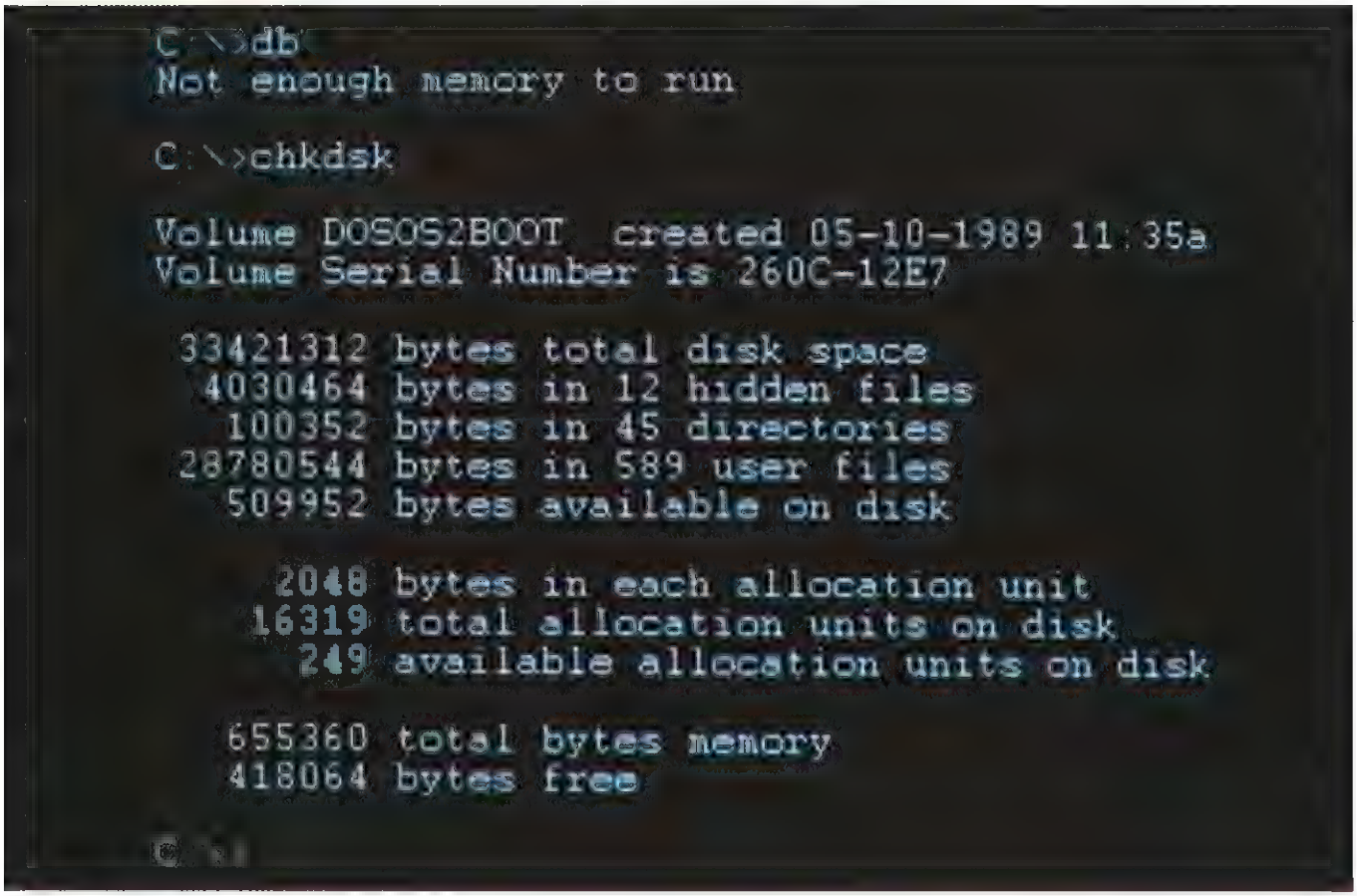
Call 1-403-448-0313
Fax 1-403-448-0315
Hours: 14:30 - 24:00 GMT

P.O. Box 5659, Station L, Edmonton, Alberta, Canada T6C 4G1

CIRCLE NO. 682

THE HARD WAY.

Have you ever tried to run a program and run into this?



THE MICROSOFT WAY.

With Microsoft Windows 3.0, you won't have to put up with the 640kb memory barrier ever again.

To find out how Microsoft Windows 3.0 breaks the 640kb barrier and increases your productivity, phone **081-879 7979**.

Microsoft®

© 1991 MICROSOFT CORPORATION ALL RIGHTS RESERVED. MICROSOFT AND THE MICROSOFT LOGO ARE REGISTERED TRADEMARKS WINDOWS IS A TRADEMARK

CIRCLE NO. 683

A Canadian Solution

In the DOS world, the 16-bit Watcom C compiler is known as a solid engine that delivers fast code. Will Watts has been looking at its 32-bit cousin.

Watcom is an unusual software house. For a start, it's Canadian, its roots go back over 20 years (remember the Waterloo Pascal compiler on the college computer?), and it's successfully made the transition from big computers to little ones. If Microway's NDP software (such as the C compiler I looked at last month) is characterised by that company's go-faster hardware background, then Watcom's output has a sort of 'traditional quality software since 1425' smell to it. Watcom compilers feel like carefully-built, solid, reliable products with walnut dashes and real leather trim; when you alter a compiler switch, you half expect it to go 'clunk'.

The downside of this approach is that Watcom has been slow to adopt the windows and IDEs of modern compiler tools. If you come from a Borland/Turbo, Microsoft/Quick/PWB, JPI/TopSpeed or Zortech V2.1 environment, you are going to find Watcom tools rather primitive on the user interface. The flip side of the coin is that, presumably thanks to its experience with larger machines, Watcom has for some time led the field in optimisation technology; for example, Watcom was (to my knowledge) the first to introduce register-based parameter passing in a DOS C compiler; Microsoft and JPI have followed. Keen to retain its innovative reputation, Watcom is currently working on (for release in the summer) a 32-bit C-based toolkit for Windows and something called the 'C/p16 Compacting and Optimising Compiler'. This latter is a 16-bit DOS compiler which will use interpretive techniques to reduce considerably the size of object code. It is intended that rarely-used modules can be compiled to pseudo-object code and linked with conventionally-compiled object modules of the frequently-used stuff to produce a .EXE which is both compact and fast. (This approach must be less bizarre than it sounds: Microsoft is working on a similar system.)

I went over all the option switches, looking for the accidental 'dead slow' one that I had tripped

To the matter in hand: Watcom C-386 is now in its second major version (called 'V8.0' to keep it concurrent with its 16-bit cousin). I have been looking at the Professional Edition for DOS, which adds a protected mode version of the compiler, the VIDEO source debugger and a Profiler to the tools supplied by the Standard Edition. You also need to acquire a 386 DOS extender: either Eclipse Computer Solutions OS/386 Developers Kit, or the well-known Phar Lap 386 DOS-Extender SDK. Both systems require you to buy licences for any copies of the run-time system that you distribute with your software. Watcom's documentation leans toward Phar Lap, so I used V2.2d of that extender for this article.

The crystal box

Watcom C-386 is supplied in a beautiful Perspex box, which cracks, damnit, if you don't open the packaging *very* carefully. Documentation consists of seven spiral bound manuals: *C Language Reference, C Library Reference, Graphics library Reference, Optimizing Compiler and Tools User's Guide, Linker User's Guide, VIDEO User's Guide and InfoBench Editor.* There is also a *READ Me First* booklet, which I didn't, but came to no harm.

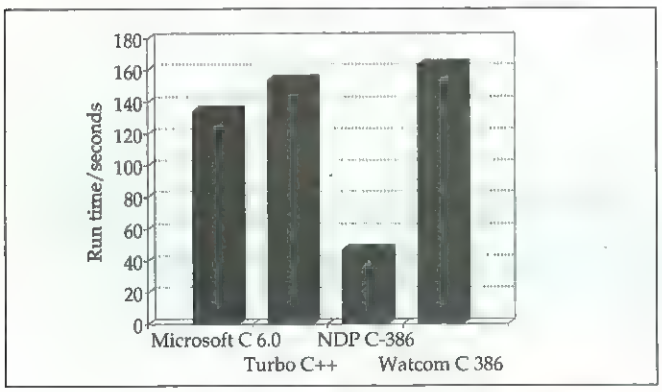


Figure 1 - Run times, including dubious NDP

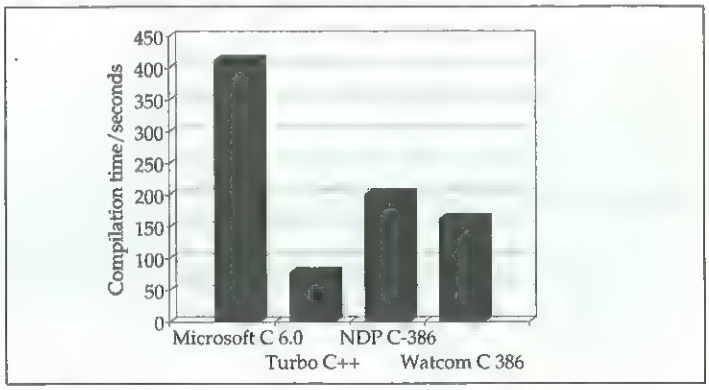


Figure 2 - Comparative compile times for Analyse

The software is supplied archived on two 1.2 MB disks which expanded to 3.6 MB on my hard disk (the complete Phar Lap toolset occupies a further 1.3 MB, but this could be reduced considerably be deleting those tools whose function is duplicated by Watcom equivalents). The INSTALL program is straightforward and unflashy, and creates a typical C compiler directory tree with separate subdirectories for binaries, libraries, includes and so on. The package includes only the 32-bit C compiler; you do not get the ordinary 16-bit DOS and OS/2 compiler, which is a pity, nor do you get the toy Express C compiler, which is no loss at all.

The compiler itself - which is command line only - is invoked by WCC386, or indirectly by WCL386; the latter is a command and link utility equivalent to Microsoft's CL. Option switches will also seem familiar to Microsoft users: for example the optimisations are turned on via a set of /o switches. The protected mode version of the compiler, supplied only with the Professional Edition of the package, is invoked as WCC386P, or by giving the /P switch to the command and link utility. As usual, C source files compile to .OBJs, which are then linked to produce .EXP files - the executable format expected by Phar Lap's RUN386 DOS extender program.

The /3r and /3s switches control the parameter passing convention. Watcom C passes its function parameters in registers by default, but there is an option switch to force it to use stack frames. There is also a complicated set of `#pragmas` which lets you set up pretty well any calling convention you like, within reason. This not only lets you do mixed language programming, it also lets you call libraries compiled for other manufacturers' Cs. A Metaware High C calling convention pragma is supplied as an example. Watcom uses these pragmas to support the `cdecl`, `fortran` and `pascal` pseudo-keywords.

Test Application

If you read my description of the Microway NDP C-386 compiler last month, you will recall that my test application consists of the calculation section of an engineering program called CADS Analyse. The CADS program performs many floating point calculations and a lot of disk I/O, see this month's *Letters* page for a critique of it as a test program. (A brief aside to address a couple of points raised in the letter. All the tests have been performed using the same floating point coprocessor, so the variance of the NDP results must be caused by the compiler. Further tests have suggested that it is the NDP software that is at fault. I chose this particular program not only to test the speed and accuracy of floating-point handling, but also to gauge the feasibility of porting a 'real' program. I'm sticking to it for the time being. Apart from anything else, I wish to retain consistency with the previous article. However, I would warmly welcome other readers' suggestions and comments, with a view to building a .EXE C compiler test suite).

Small, medium, compact and large should be mapped to flat, big, enormous and ridiculous

Porting the code to Watcom C-386 was no trouble at all, even less trouble than porting from Microsoft to Turbo C++. One reason for this is that Watcom's is the ANSI-est compiler that I've personally used (I have not tried the most recent, certificated version of TopSpeed C), passing all 79 of the Plum Hall sampler tests. Another reason is that Watcom's library contains most of the Microsoft C functions, including some graphics functions. In the end, the only code alteration that I had to do was the `#undef` and then `#define` to nothing of Watcom's `near` and `far` macros. If you think about this, this is rather surprising. Surely, I hear you say, one of the great benefits of a 32-bit compiler is that it uses a flat memory model, with 4 GB of address space into which all code and data is loaded.

This is exactly right. Watcom, however, is already worrying about the time when 4 GB won't be enough; so it has equipped the compiler to produce all the small/medium/compact/large memory models that you get with a 16-bit compiler; the difference being that a `far` pointer contains 48-bits, which gives you ample of space, just as long as you can squeeze your arrays into 4 GB each. Watcom's libraries currently don't support anything other than the flat/small model. Incidentally, wouldn't it be a good idea to devise an alternative nomenclature for these 32/48-bit memory models, to avoid confusion with their baby cousins? I suggest that small, medium, compact and large should be mapped to *flat, big, enormous* and *ridiculous*.

After minimal preparation, then, I had the test program up and working. Figures 1 and 2 give the run and compile times for Watcom, together with the compilers tested last month. In all cases, the compilers had all their optimisation switches on full. Watcom's compiler, on which I would have laid real money on coming in first, produced the slowest code of all.

This discovery was the equivalent psychological experience of looking up a favourite film in *Halliwell's Film Guide*, and finding that it had no stars and a dismissive comment. I didn't believe it. I went over all the option switches in the make file, looking for the accidental 'dead slow' one that I had tripped. It wasn't there. I phoned Watcom. The technical support bloke didn't believe it either. We went through all the switches that I had set. Everything was in order. He muttered that I/O intensive applications run more slowly under DOS extenders, because of the constant need to switch modes. True enough, but it didn't bother the Microway offering.

As you will have deduced, we have not yet found anything amiss; so Watcom's poor timing must stand. I still don't believe it, and remain in dialogue with Watcom to try to track the problem down. If anything happens, I'll let you know. By the way, Watcom got the answer right:- there were no significant differences between Watcom and Microsoft/Turbo C++ results.

Libraries

As mentioned before, Watcom's libraries are modelled closely on the Microsoft 5.1 set, and are exactly the same as those supplied with Watcom's 16-bit offering (in fact, the same manuals are supplied with both, with the occasional difference noted, such as the return of 32-bit registers from the DOS and BIOS functions). Microsoft C V6.0 extensions do not yet seem to have been included; for example, C 6.0 offers 114 graphics functions, Watcom provides just 53. Most of the missing functions are either text font handling routines (Watcom's facilities are thin in this area) or the business graphics calls, which draw pie charts, scatter graphs and so on. There is support for the CGA through to VGA graphics standards, including Hercules; but nothing for the IBM 8514/A or the various Super-VGA cards.

The library functions are contained in seven .LIB files. The main function library calls are duplicated in CLIB3R.LIB and CLIB3S.LIB,

allowing them to be invoked from programs compiled with either register- or stack-based calling conventions. Similar doubling-up occurs in the maths libraries. There are two basic sets of code; MATH387 assumes floating point instructions are generated inline, and provides an emulator (which need not be linked in if a coprocessor is a certainty) to handle FPU instructions, MATH3 - which can also use a coproc if one is detected - requires the compiler to do its arithmetic via a series of calls. There is no Weitech support. The graphics functions are held in GRAPH386.LIB, and are only supplied in 'register' mode.

Other Tools

A major plus point for this package is that there is no skimping with the add-on tools or their documentation. There is a fully-fledged make utility, called WMAKE, which seems to have more than its allotted share of options and twiddles; for example, macros to pull out the first and last of a dependency list, a heuristic algorithm for path searching and a '::' operator that lets you set up some dependency relationships that cannot be described to lesser MAKEs. It also has the occasional non-standard feature. For example, WMAKE wants the line-extension character to be & not \, which gave me a little trouble initially. Watcom justifies this by pointing out that MAKE lines that read

```
CD \
```

are ambiguous.

Unusually, the C compiler does not have an option switch to cause it to generate assembly language. Instead, Watcom provides an object file disassembler which uses debug line number information to include the relevant section of original source code as a comment to the assembly listing.

Phar Lap supplies a linker and a librarian with its DOS Extender kit, but you don't need them. WLINK the linker is a most satisfactorily versatile program. As well as supporting the Phar Lap and Eclipse 386 linking standards, it functions as an ordinary DOS linker, a Windows and OS/2 linker (including DLLs), produces Novell NetWare NLMs and can generate executables for the QNX operating system. The linker has a command line format all of its own - you'll need to alter all your response files but there is a utility, MS2WLINK , to do the conversion - but it is logical and consistent. WLINK comes paired with a librarian WLIB which can handle the same variety of object files.

The debugger, VIDEO, and the Execution profiler are privileged add-ons for Professional Edition owners only. Both are blessed with text windows + mouse style user interfaces, which in both cases are just a little quirky. The debugger is a bit like CodeView in that, despite all the windows, one is still very much aware of the command line interface underneath. This means that it takes a bit of getting into. Once there, you'll find a rich set of options and features. There are expression handlers for FORTRAN and C, all the usual watch and break point facilities, a customisable menu and an impressive set of remote debugging facilities. Remote debugging is when you run the test program on one machine and the debugger interface on another. VIDEO gives you a wide choice of hardware to join the two components: via serial ports, parallel ports, across Novell or NetBIOS networks, and even, rather surprisingly, between two DESQview windows running on the same machine.

As a prelude to running the profiler, you must first compile the test program with the debug info switch, then run it with the execution sampler. This latter program uses the clock interrupt to spring to life at regular intervals and determine the current position of the instruction pointer in your code. This information is recorded in a file. You can also place 'marks' in your source code, delimiting routines of special interest. When one of these is encountered during sampled execution, the event is also recorded in the file.

The profiler accepts this file as its input, and uses it to draw a sort of histogram of the percentages of execution time taken up by the various components of your program. When you have identified the slow bit, you can 'zoom in' down to the module concerned, then the particular block of C code, and finally down to assembly language instructions. The business of letting you scurry around your source files searching for the blockage is handled very cleverly - this tool is much easier to use than some of the earlier, non-interactive profilers you may have encountered.

I'm running out of space here, but a quick mention for text editor, which is quite simple, and vaguely CUA-ish, and loads better than my recollection of the one which used to come with Express C. It supports a mouse, simple cut and paste, multiple file buffers (using a horizontal split-screen arrangement that looks a bit primitive in these days of movable, resizable windows) and keystroke recording. If you have already wedded the editor of your choice then I doubt that this one will cause marital strife; if you are writing programs with an out-of-date edition of some fat, slow-loading word processor, then this just might be the ticket.

Documentation

Documentation is just to my taste. My taste requires that there should be many, smallish manuals, that these should be properly bound and printed (not, for example, in large; ring binders that constantly spring open), and that all information is printed out on paper - no dependence on online help databases to find the compiler switches. Actually, Watcom provides *no* online documentation at all - but as it does not supply an IDE in this package, this is no surprise. Last time I looked at a Watcom product, I commented that the documentation suffered from 'the guy that wrote the program wrote the manual' syndrome. It seems to me that this phenomenon has been largely sorted out: the text is generally clear and concise. One thing I happened to notice - a sure sign of a classy technical writing or Latin training this - is that the authors remember to use 'datum' as the singular of 'data'. Reassuring, then, to find even a small cock-up: the word 'Break' appears twice (before and after 'break point') in the index of the VIDEO manual - perhaps Watcom should have a glance under the bonnet of its index sorting program.

Conclusion

You may have detected that I am reluctant to slam Watcom for its poor speed showing. The fact is, aside from this one problem, the software was a pleasure to use. However, the compiler is marketed with an emphasis on performance. If my test code has fallen through a hole in the optimiser, then it's a big hole. Of course, the code could certainly be made to run faster by, for example, taking advantage of the multi-megabyte address space made available by the 32-bit architecture to reduce disk I/O. But this violates the rules of my original game, which was to discover the benefit of porting a DOS 16-bit C application, with minimum alteration, to a supposedly intrinsically faster environment.

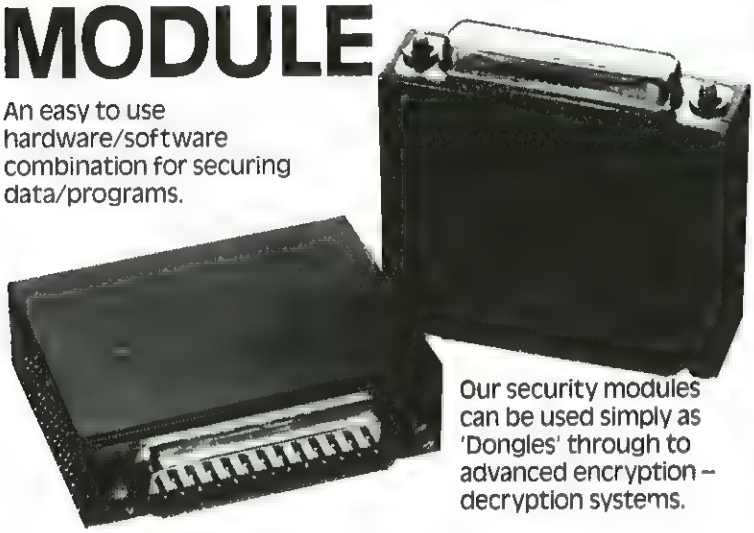
In the testing of two 386 C compilers, I have found one which gives inaccurate results, and one which generates unexpectedly slow code. Hmm. There seems to be more to these big compilers than meets the eye - and there are many more to come.

EXE

Watcom is distributed in the UK by Grey Matter (0364 53499). Watcom 386 C Standard costs £515, Watcom 386 C Professional costs £680 and the Phar Lap 386 DOS-Extender Software Development Kit is priced at £370. All prices sans VAT.

SOFTWARE SECURITY MODULE

An easy to use hardware/software combination for securing data/programs.



Our security modules can be used simply as 'Dongles' through to advanced encryption – decryption systems.

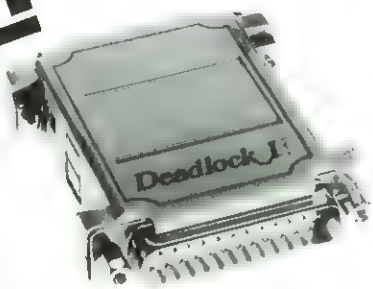
- ★ For IBM PC family (and compatibles)
- ★ Variants for other systems
- ★ Advanced construction – rugged and compact
- ★ Proven throughout the world
- ★ Transparent operation – user friendly
- ★ Driver software supplied for variety of applications

Control Telemetry of London
11 Canfield Place, London NW6 3BT
Tel: 071-328 1155 Fax: 071-328 9149

CIRCLE NO. 685

Deadlock-I

ONLY £7.50



1. Software protection units that are programmable by the Software House.
2. No need to keep a stock.
3. Ten (10) days delivery guaranteed.
4. The codes can be changed if desired by the software house.
5. Supplied with **FREE** software.
6. Are you already using DONGLES? You can have us as a second source. (Call for details).

Please send all orders to B. L. Computer Security, 101 Hendon Lane, Finchley, London N3 3SH, or call us on 081-343 0734 or Fax us on 081-346 2672.

CIRCLE NO. 685

System Science

C COMPILERS

Microsoft C 6.0 with PWB	£245.00
Quick C & Quick Asm 2.51	£99.00
Quick C ver 2.5	£65.00
Borland C++	£245.00
Zortech C++ Dev Ed	£275.00
Watcom C 8.0	£255.00
Aztec C86 Comm	£265.00
C compilers for 386	£call

C GRAPHICS & DATAFILE

Halo	£195.00
GSS GDT	£445.00
Metawindows	£175.00
Essential Graphics (no roy)	£275.00
PCX Prog Toolkit (new 4.0)	£165.00

C DATA & MISC

Btrieve for DOS	£345.00
XQL for Btrieve	£475.00
Ctree (source)	£195.00
CodeBase 4.2 (source)	£175.00
db_VISTA III	£545.00
GL Financial Mathlib	£245.00
Blaise C Tools Plus	£89.00

C SCREENS & COMMS

Cscape, Look & Feel	£355.00
Vermont Views & Designer	£295.00
Panel Plus II	£275.00
C Worthy & cwARCHITECT	£call
Essential Comms (src)	£195.00
Blaise C Aynch Manager	£115.00
Greenleaf CommLib	£185.00

PROGRAMMERS TOOLS

SPE prog editor	£189.00
MKS Toolkit (Unix toolset)	£155.00
PVCS Prof (lower price)	£375.00
PolyMake	£135.00
vsDesigner (Case Tool)	£475.00
Brief Editor 3.0	£195.00
Kedit Editor	£110.00
Flow Charting 3	£175.00
Dan Bricklin DEMO II (new)	£195.00
MS Progs Lib 1.2 CD	£225.00
Personal Rexx	£110.00

ASSEMBLERS & LINKERS

MS Macro ASM 5.1	£90.00
Turbo Debugger & ASM 2.0	£99.00
RTLink Plus	£345.00
Plink 86 Plus	£295.00
Sourcer & Bios (DisAsm)	£125.00

CROSS DEVELOPMENT

Introl C or Modula-2	£1445.00
Aztec C Cross	£275.00
IAR C Cross with CSpy	£call
2500AD Cross AsM	£135.00
Simulators & others	£call

FORTRAN

MS Fortran-77 5.0	£235.00
Lahey F77L-EM/32 & 386 OS	£795.00
Salford FTN77/386	£765.00
Lahey Personal Fortran	£79.00
RM Fortran Forte	£475.00

PASCAL, BASIC

Turbo Pascal (6.0)	£75.00
Object Professional (for TP)	£110.00
MS Quick Pascal	£60.00
Blaise Power Tools Plus	£89.00
Blaise Asynch Plus	£89.00
MS Basic 7.1 with PWB	£255.00
MS Quick Basic 4.5	£65.00
PowerBasic (ex Turbo Basic)	£85.00
many Pascal, Basic libraries & tools	£call

OTHER LANGUAGES

Smalltalk V/Windows	£295.00
TopSpeed Modula-2	£145.00
MS Cobol 4.0 (new ver)	£445.00
RM Cobol-85 ver 5.0	£995.00
LM1 or Forth	£225.00
PC Logo	£50.00
Clipper 5.0	£395.00

MEMORY, MATHS etc

386 Max 5.0	£75.00
QEMM/386 5.1	£60.00
QRAM for 286	£55.00
Above Disc (LIM Emul)	£69.00
Norton Utilities (new ver 5)	£110.00
Spin Rite II	£69.00
Derive (Symbol Alg) 2.0	£135.00
MicroStat II	£295.00
Mathematica by Wolfram	£call

WINDOWS 3.0

MS Windows 3.0	£85.00
MS Windows SDK & MS-C	£425.00
MS Windows SDK 3.0	£325.00
Actor 3.0 with Lang Ext	£625.00
WinPro/3 from Xian	£525.00
WindowsMaker	£595.00
Toolbook by Asymetrix	£295.00

"OUR RANGE INCLUDES MANY MORE ITEMS FOR DOS, OS/2, UNIX ETC. PLEASE ENQUIRE"

3-5 Cynthia Street, London N1 9LF
Fax: 071-837 6411

BARCLAYCARD VISA Access

Prices are subject to change and alteration Prices include UK Delivery but are exclusive of VAT

071-833 1022

CIRCLE NO. 686

The European AI Language with the American Name

PROLOG is a programming language based explicitly on formal logic, favoured by researchers in Artificial Intelligence. Dave Green investigates why it thinks it's so special.

They say that the language-name SNOBOL derived from its writers concluding that it didn't stand a 'snowball's chance in hell'. PROLOG, on the other hand, is widely regarded as having a much better chance of surviving a trip to mythical Hades, because that's where most people think it came from. PROLOG, more than any other computer language I've ever encountered, fits the bill as *the* mad computer language from Hell. Remember that feeling of terror and confusion when you saw your first OOP program or first had to debug extensive and uncommented C source code? Well, you're about to experience that same sensation once again, as I ask you to put aside almost everything you ever learned about procedural programming, and enter the mad mad declarative world of PROLOG.

A Scottish Evil

There's always been a fair degree of rivalry between PROLOG and LISP regarding which should be considered the language of choice for AI applications. LISP is the older of the two, having been devised in the mid-1950s by one of the American founders of the field of Artificial Intelligence, John McCarthy. On the other hand, it's difficult to trace the origins of PROLOG back much further than the late 60s and early 70s, when the idea of programming languages based explicitly on formal logical systems was being proposed by, among others, Kowalski and Hayes.

The honour of writing the first ever PROLOG program is credited to a team working in Marseilles on a natural language comprehension project in the early 1970s. However, the development of the language was due to the work of David Warren and Fernando Pereira at the Department of Artificial Intelligence at Edinburgh University. They were responsible for defining an early standard of the language, known as the Edinburgh syntax. Worldwide interest in PROLOG was kindled by the Japanese announcing that they intended to use it as the language for their Fifth Generation Computing Project. Although we don't hear so much about the Fifth Generation Project nowadays, enthusiasm for PROLOG as a symbolic high-level language continues unabated, with interpreter/compiler packages available for a variety of machines.

Predication

So what makes logic-based programming so different from the 'normal' sort? The best way to illustrate this is to give a flavour of what a PROLOG program looks like, so take a look at the listing of the standard .EXE Triangle program rendered in PROLOG in Figure 1. And then try and figure out how it works. Confusing, isn't it? While there's a superficial resemblance to function-based languages like LISP, it's by no means immediately apparent in PROLOG how arguments are passed back and forth, or indeed why certain functions (such as `input` and `displ` in this example) have been defined twice (if not more). And how come there doesn't seem to be any distinction between program and data? And for that matter, where have all the variables gone?

PROLOG programs do not consist of instructions in the conventional sense. Instead they feature what are called 'terms', which more or less correspond to logical statements about what the program effectively knows. A good example of a simple term can be seen in the first line of the Triangle program:
`number_of_sides(triangle, 3).`
This is a term (also called a predicate) which broadly corresponds to the statement that triangles have three sides and that three-sided shapes are triangles. To be more precise, it states that a possible solution (and indeed the only solution currently available) for the predicate `number_of_sides(X, N)` is `X =triangle, N = 3`.

```
%  Triangle program in PROLOG for .EXE
%  Author:        Dave Green, Mar. 1991

% Call from top level with triangle(X).

number_of_sides(triangle, 3).

triangle(Type) :-
number_of_sides(triangle, N),

write('Enter '), write(N),
write(' lengths, with .[N/L] after each'), nl,

input(N, Sides, []),
write('Triangle entered: '),
displ(Sides), nl,
!,
type(Sides, Type),
nl, write('This is a triangle of type: '),
write(Type), nl.

input(N, Current, Current) :-
length(Current, N).

input(N, End, Current) :-
        \+ length(Current, N),
%%% ('\+' is SICStus 'NOT' operator)
        read(Value),
        input(N, End, [Value | Current]).

displ([]).

displ([Head | Tail]) :-
displ(Tail),
write(Head),
write(' ').

type([X,Y,Z], illegal) :-
X >= Y + Z;
Y >= X + Z;
Z >= X + Y.

type([X,X,X], equilateral).

type([X,X,Y], isosceles).

type([Y,X,X], isosceles).

type([X,Y,X], isosceles).

type([X,Y,Z], scalene).
```

Figure 1 - The Triangle program in SICStus PROLOG

There are many ways to boost your software development efforts

A more sensible way of launching your next software development project is with Enfin/2, a new 4th Generation Language software development tool which allows you to produce complex software applications quicker and more easily than you had previously thought possible.

It lets you create applications for either Microsoft Windows 3 or OS/2 Presentation Manager Graphical User Interface (GUI), with built-in graphics, charts, data storage, SQL database, sophisticated graphical reports and communication links to the outside world. Enfin applications developed in one environment will run in the other.

Enfin/2 lets you build complex systems out of simple building blocks, or 'objects'. It supplies nearly 150 standard objects to give you a valuable head start, allowing you to prototype, explore your business requirements and come up with solutions for your business' changing needs.

But things don't stop there.

There is a sophisticated range of code generating editors within Enfin/2 to increase productivity, whilst an optional compiler produces standalone, royalty-free applications to use on as many workstations as you like.

Enfin/2 is backed by QA, the UK's premier PC technical training and products company, who will train, consult and support your efforts. Contact us for more details, or to book onto one of our free introductory seminars.

ENFIN/2

4 GL Object Oriented Development Environment

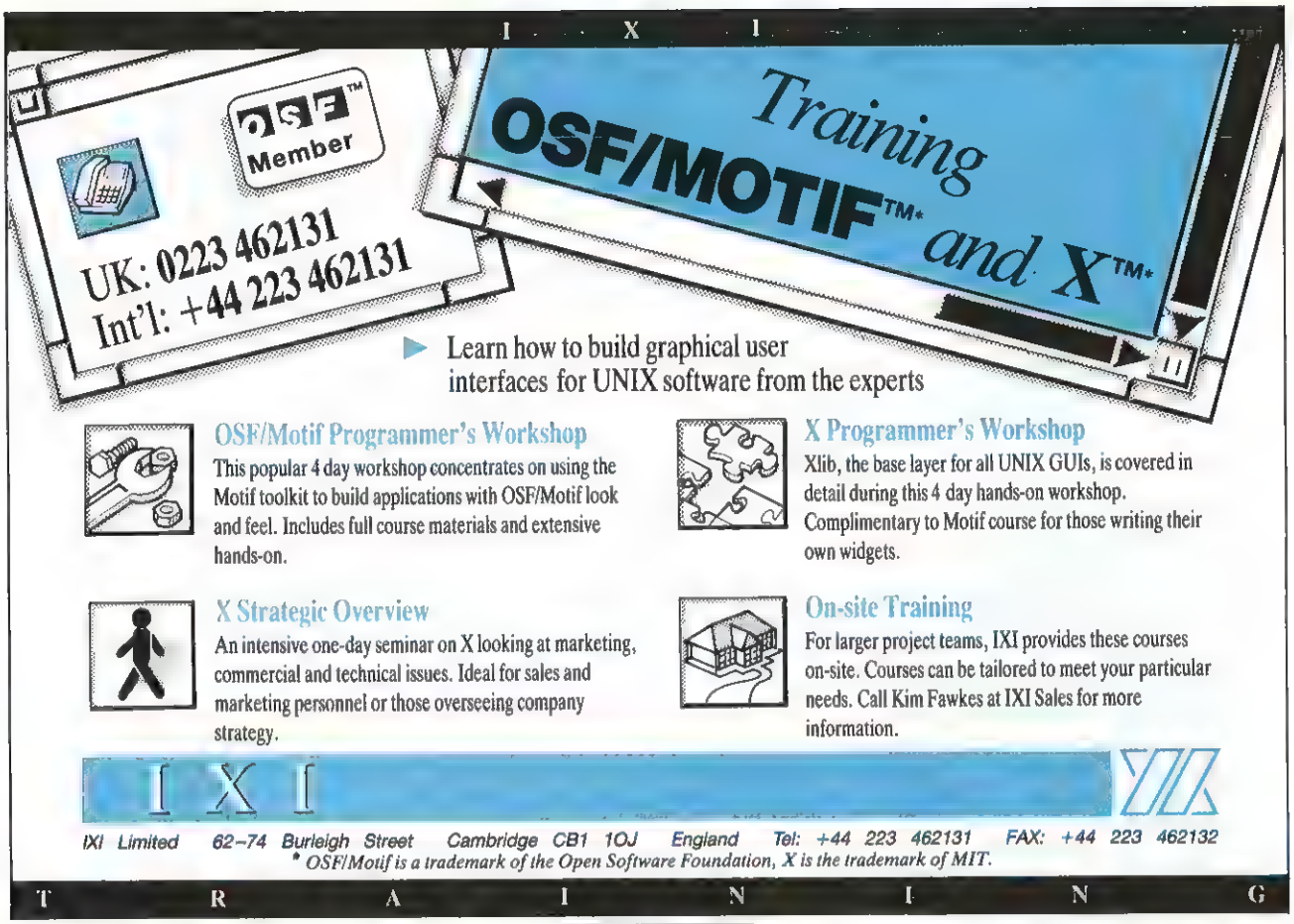
Tick here for full details: ENFIN/2 for OS/2 ☐ ENFIN/2 for Windows ☐ Introductory Seminars ☐ ENFIN/2 Training ☐

NAME	TITLE
COMPANY	
ADDRESS	
	POSTCODE / TEL NO

EXE 4/91

Mail to The Products Group, QA Training Ltd., Cecily Hill Castle, Cirencester, Gloucestershire GL7 2EF. UK.
Tel. 0285 655888 Fax. 0285 640181

CIRCLE NO. 687



CIRCLE NO. 712

EUROPEAN INFORMIX USERS GROUP

Informix is a trademark of Informix Software Inc.

Forthcoming Seminars to be held in Central London

16th May 1991	*Interoperability*
12th Sept 1991	*Multimedia systems*
5th Dec 1991	*Complex databases*

For further details contact:

Bryce Campbell (0532) 738 244
Jeremy Russel (081) 446 6481
Jerry Nichols (071) 251 2128

CIRCLE NO. 696

MULTIPLE INTELLIGENT COM PORTS FOR OS/2

THE IMPAC 8 & 16 INTELLIGENT COMMS CARDS

* MICROSOFT COM X COMPATIBLE
* 8 TO 64 FULL MODEM PORTS
* REDUCE CPU OVERHEAD FROM 40% TO 1% PER PORT
* FITS ALL ISA (PC/XT/AT) MACHINES
* DOES NOT USE SHARED RAM
* PROVEN FIELD PERFORMANCE
* UK DESIGN AND MANUFACTURE
* DRIVERS FOR DOS, OS/2, UNIX & XENIX

DEALER ENQUIRIES WELCOME

For more information contact:
SCL Downswood, Carlton Road
Reigate, Surrey RH2 0JQ
Telephone: 0737-762200

CDOS DRIVER ANNOUNCED

CIRCLE NO. 688

PROLOG programs are comprised of a number of these terms stored in an order-sensitive database - 'running' a program results in the interpreter searching through this database in order to obtain the information, by means of a simple pattern-matching process. This can be demonstrated in PROLOG's interactive mode by typing in terms, and seeing whether you get a 'yes' in response (meaning that the term could be successfully proved from the database) or a 'no', meaning that it could not.

From the sample interaction shown in Figure 2, you can see that it is possible for the system to prove to its own satisfaction that triangles have three sides, and not two. And no less surprisingly, PROLOG also responds 'no' to the assertion that squares have four sides, as it will whenever it is faced with a statement which it is unable to prove from its database, irrespective of whether it happens to be true in the real world or not.

Now, suppose you wanted to make some rather more substantial enquiries about the potentially earth-shattering information the database now contains on how many sides a triangle has. To do this, you can just type in incomplete predicates with variables in the appropriate places, and PROLOG then gives you all the solutions it can find. PROLOG variables are traditionally distinguished from atoms like 'triangle' or '3' by starting them with either an upper case letter or an underscore character; as you can see from Figure 2, this approach can be used to find out how many sides a triangle has or, conversely, which shapes the system knows that have three sides, or even all known solutions to the number_of_sides predicate.

After printing out a solution, PROLOG programs wait for user input on whether to continue searching for others - hit RETURN at this point and execution will terminate with the 'yes' response indicating success. Pressing ';' will result in the search persisting for other solutions, 'no' in the example case shown meaning that no others were found.

It's a goal!

I've spent quite a lot of time explaining the basics, but that's because once you understand them, that's practically all there is to PROLOG - things just get more complicated when you start using the same logical terms to implement programmed procedures. By calling the Triangle program at top-level with triangle(X) in a PROLOG environment you are asking the program to find a value of X which satisfies the definition of the predicate triangle in the database.

However, triangle(Type) is not defined in the simple way that number_of_sides is; the ':-' operator determines that triangle(X) can only be solved if the various goals to the right of the operator can also be solved for that value of X. Successful solutions for both sides of the clause are dependent on giving appropriate values to all variables in all of the expressions given.

Now, since PROLOG attempts to satisfy sub-goals in the order in which they are found in the database, separated by either commas (representing the AND operation) or semicolons (representing OR), it now becomes possible to implement a form of procedural programming by linking together logical terms. This is how most of the predicates in the Triangle program work, and partly explains the similarity to function-based programming languages, PROLOG programs simply appearing to call predicates in a given order as opposed to calling functions.

While the difference between predicate- and function-based programming is a subtle one, it is also quite important - it is not just the case that PROLOG predicates are just LISP functions only with extra arguments. PROLOG predicates can return any number of output values, and, as we saw for the number_of_sides term, can often work in more than one direction, effectively using the arguments given as constraints defining the nature of all the solutions to be found. In this respect at least they can be considered more powerful than the simple single-value functions.

If at first...

Furthermore, LISP functions are often forced to use their single output values to indicate when they have failed, typically returning NIL. However, flow of control in PROLOG programs is kept separate from the returned values, and is defined in terms of whether a given goal has succeeded or failed. Success can mean that a solution can be found in the database for a given combination of term and variables, or, in the case of some built-in PROLOG predicates such as write, that their associated process has been carried out. Unlike LISP, the failure of a PROLOG predicate is not explicitly returned as a value.

Instead, failure to satisfy goals or sub-goals is used by the interpreter as an indication that it should backtrack to the last choice point it encountered in the program's execution, and attempt to obtain a successful solution by using any alternative options it could have chosen at that stage. For instance, adding an additional term number_of_sides(triangle, 4) to the Triangle program as given here would result in a choice point being set for the first sub-goal of the predicate triangle(Type) on execution; subsequent failure of other sub-goals would have eventually resulted in backtracking to this point and the interpreter retrying all the sub-goals with the different value of N = 4.

```
| ?- [tri].
{consulting /uhuh/prolog/tri.pl...}
{tri consulted, 259 msec 2575 bytes}

yes
| ?- number_of_sides(triangle, 3).

yes
| ?- number_of_sides(triangle, 2).

no
| ?- number_of_sides(square, 3).

no
| ?- number_of_sides(square, 4).

no
| ?- number_of_sides(triangle, N).

N = 3 ?

yes
| ?- number_of_sides( shape, 3).

_shape = triangle ?

yes
| ?- number_of_sides(X, N).

N = 3,
X = triangle ? ;

no
```

```
| ?- triangle(X).
Enter 3 lengths, with .[N/L] after each
|: 3.
|: 4.
|: 5.
Triangle entered: 3 4 5

This is a triangle of type: scalene

X = scalene ?

yes
| ?- triangle(X).
Enter 3 lengths, with .[N/L] after each
.: 1.
|: 4.
|: 4.
Triangle entered: 1 4 4

This is a triangle of type: isosceles

X = isosceles ? ;

This is a triangle of type: scalene

X = scalene ? ;

no
| ?- halt.

{ End of SICStus execution, user time
0.460 }
```

Figure 2 - Sample interaction with the Triangle program and its sub-predicates

```
employee(babbage, 20, [engineering]).
employee(einstein,30, [astrophysics, fast_cars]).
employee(edison,  15, [inventions, fast_women, loose_cars]).

earns(Employee, Salary) :-
employee(Employee, Salary, _).

has_2_hobbies(Employee, Hobby1, Hobby2) :-
employee(Employee, _, [ Hobby1, Hobby2 ]).
```

Figure 3 - Toy database using predicates as structured data

Of course choice points also occur at those stages in the program where predicates are defined more than once; while PROLOG automatically tries the earliest predicate definition first, should that one fail, it will then backtrack to that choice point to see if it has more success with any subsequent versions. This explains why there are two separate definitions of both `input` and `displ` in the Triangle program, each of which will apply under different calling conditions.

Forced backtracking to find other solutions is also what happens when the user presses ';' after PROLOG has offered a possible solution. In other words, a fully intensive run of any PROLOG program requires not only that some solutions succeed, but that all other possible solutions must fail. It is not known what effect this may have had on the world-view of veteran PROLOG programmers; we can only hope that it does not entirely counteract the gritty optimism of the PROLOG popular proverb - 'if at first you don't succeed, backtrack to all previous choice-points and continue with exhaustive depth-first search'.

There are some circumstances in which you don't want backtracking, so PROLOG conveniently gives you the '!' (or 'cut') operator, beyond which this process will not occur. This operator is used in the main `triangle(Type)` predicate to limit the solutions given to those which apply to the original input values.

Further Lunacy

The .EXE specification of the Triangle problem suggests that inputting and displaying the lengths of the sides offers the ideal opportunity to demonstrate the 'looping idiosyncrasies' of the language under scrutiny. What's most idiosyncratic about PROLOG is that there aren't any loop constructs, at least not in the traditionally iterative FOR/NEXT-type way. And so the art of devising recursive functions has been developed by PROLOG programmers to almost its logical extreme, although fortunately this is not so wasteful of stack space as you might expect since PROLOG compilers have built-in optimisers which are specifically designed to compensate. Also note the subtly different recursive structure of the predicates `input` and `displ`, required to print out the list of the triangle lengths in the same order in which they were input.

'Now wait a minute,' I hear you protest. 'If there's no iterative looping constructs in PROLOG - then how do you access arrays?'. This turns out not to be a problem for PROLOG, since it does not have arrays. Probably the most complex data structure made available by PROLOG is the list, represented by items separated by commas enclosed within square brackets. Although not as list-based as its complementary language LISP, which uses them for both data and program structures, PROLOG's use of lists once again highlights how it is intended to be a high-level language operating on symbolic information rather than mathematical formulæ. FORTRAN it most certainly is not.

All direct manipulation of lists is carried out using the '|' primitive, which separates the list's head from its tail. Given the recursion-based nature of the language, this is usually all you need, and it is quite typical of PROLOG systems to limit the number of built-in functions provided to the barest minimum - Edinburgh's SICStus system, for instance, doesn't even give you predicates to join lists together or check whether an item is a member of a list or not. While LISP packages usually come with extensive libraries of functions to do all this and more, it does seem that PROLOG has remained a smaller language in this respect.

In any case, it is doubtful how much use arrays in PROLOG would be, since, as you will have realised by now, the language has no global variables either. Fortunately this can be overcome by using data structures which are globally accessible - logical terms held in the rulebase. Figure 3 gives a short toy example of how you might construct a database for an imaginary company in the habit of employing dead scientists, complete with predicates for accessing details of their salaries and for identifying those individuals with exactly the right number of interests outside the company. PROLOG allows a number of ways of manipulating predicates as arguments, and can add and remove them selectively in the database from within the same program. This also makes it possible to write meta-predicates which can act on predicates and goals themselves, making it relatively easy to write meta-interpreters for modified versions of PROLOG or simple PROLOG-based expert system shells.

In Conclusion

The whole point of logic-based languages is to produce declarative programs which represent problems in terms of their higher-level constraints, as opposed to listing the procedure which will solve them. As can be seen from the `type` predicate in the Triangle program, PROLOG can express such things simply and economically. Unfortunately, implementing even straightforward programming procedures using the same approach can sometimes produce listings which are utterly inscrutable to anyone used to more conventional languages.

Presumably there is some sort of sinister trade-off at work here - within PROLOG it is almost trivial to write short programs to perform tasks that would take many more lines of code to implement in C, because things like pattern-matching, search strategies and even a basic grammar parser are all provided. And part of the charm of programming in PROLOG lies in trying to think in a qualitatively different way to what you've been used to when programming in other languages. After all, the idea behind backtracking isn't too hard to cope with; it's just coming to terms with it as the single control structure in your programs that's difficult. Possibly the most remarkable aspect of PROLOG as a programming language is that everything is so extraordinarily simple. Apart from the programs, of course.

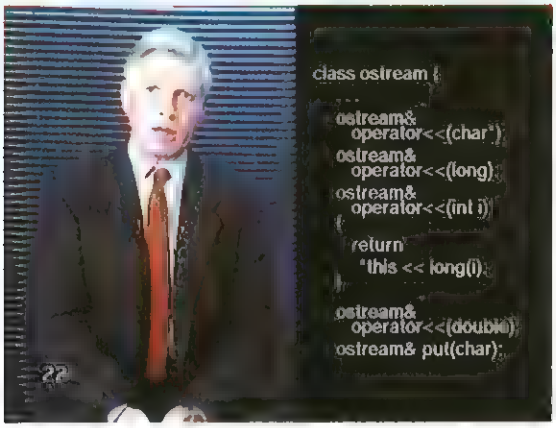
EXE

Dave Green is the name of a Natural Language Front End currently being beta-tested at the Department of Artificial Intelligence, University of Edinburgh. Its areas of specialist domain knowledge include the application of psychological research data to programming problems. Communicate with this product via JANET at davidg@aipna.ed.ac.uk but please - no overlong sentences or embedded clauses.

If you are interested in reading further, it recommends Prolog Programming for Artificial Intelligence, 2nd Edition (Author: Ivan Bratko, Publisher: Addison-Wesley, ISBN: 0-201-41606-9). This contains more than you'll ever want to know about AI applications and PROLOG.

VIDEO COURSE

Learn C++



NO RUSH!
NO TRAVEL!
NO HOTELS!
ONLY £299.95

Learn C++ Now!
The great exodus of programmers from C to C++ has begun! Since C++ builds on C, it's the easiest OOP language to learn. That's why it's called "the language of the 90's".

Why the rush?
PRODUCTIVITY! Yes, C++ programmers can write programs in less time requiring less maintenance. Large projects become much easier to manage.

Unfortunately, learning C++ can be very costly. Classroom instruction is expensive even without the travel and hotel costs. Of course, not learning will cost you even more in the long run. Now there is an alternative!

The top C++ video tutorial at the lowest possible price.
The C++ video tutorial from Zortech is the ultimate C++ training tool for work or home at only £299.95. It comes on six VHS video tapes containing 32 clear, extensive tutorials.

Used in conjunction with the concise workbook and tutorial disk, you will find everything you need for fast-track C++ tuition.

The course is generic (i.e. compiler and hardware independent) and is available with or without the award winning Zortech C++ Compiler for MS-DOS and OS/2.

Start writing C++ code within a week.
As a C programmer, you will start producing C++ code within a week of concentrated use of this course. Alternatively, spend just an hour a day watching the video and working through the suggested exercises to learn C++ in only six weeks!

The Leader in video tutorials.
If you don't already know C, you can join tens of thousands of programmers world-wide who have learnt C with the Zortech Complete C Video Course, described as:

"An excellent bargain ...
... I heartily recommend"
Gary Ray, PC WEEK

Save your company thousands of £££'s.
One programmer can train for only £299.95, but you can train ten programmers for just:

C++ Course	£299.95
9 Extra Workbooks	£179.55
Total 10 students	£479.50

Yes! Only £47.95 each! (With all the FREE refresher courses you need!)

Now, Zortech with its new C++ Video Tutorial has refined the art of video tuition and presentation even further for the 1990's.

Only £299.95 complete
- *Six Videos with 32 lessons*
- *256 page workbook*
- *Tutorial disk*
- *Compiler & hardware independent*
- *NTSC or PAL format*
- *Tax deductible*

Don't delay, order now!
Just mail the coupon or call the order hotline for same day shipment.

USA: Zortech Inc.,
1165 Massachusetts Avenue,
ARLINGTON, MA02174
Voice: 617-646-6703
Fax: 617-643-7969

EUROPE: Zortech Ltd.,
106-108 Powis Street, LONDON
SE18 6LU
Voice: 44- 1-316-7777
Fax: 44- 1-316-4138

IN EUROPE CALL:

44- 1-316-7777

ORDER FORM
Please rush me these items:

Qty	Description	Price
___	C++ Video Tutorial	£299.95
___	Extra C++ Workbooks	£. 19.95
___	Zortech C++ Compiler	£129.95
___	Developer's Edition	£299.95

Please add £5 + VAT postage in UK
All UK orders please add 15% VAT
International shipping charged at cost

Name _____
Company _____
Address _____
_____
_____
Phone _____
MC,VISA or CHQ_____ Expiry ____
Card No. _____

EXE 4 91

CIRCLE NO. 689

WE GUARANTEE Y
WIDER RANGE OF SO

Over 10,000 Software Products

● COMMUNICATIONS ● DATABASES ● dBASE ADD-ON'S ● DEBUGGERS ● DEMO'S & PROTOTYPING ● DESKTOP PUBLISHING ● DIAGNOSTICS ● DOCUMENTORS ● E-MAIL ● EDITORS ● EDUC

LOTUS 1-2-3 ● 386/486 PRODUCTS ● ACCOUNTING ● C LANGUAGE ● CAD/CAM ● CASE & FLOW ● CD-ROM ● CLIP-APT ● CLIPPER

@Base	145.00	Close-Up/Customer	139.00	dSalvage Professional	149.00	Greenleaf Communications	239.00	LotusWorks	
386MAX Pro	79.00	Close-Up/Support	169.00	Dungeon Master	24.99	Greenleaf SuperFunctions	179.00	Mace Utilities '90	
ABC Flowcharter/Windows	199.00	CO/Session	159.00	Duplicator Toolkit	69.00	Gray F/X	299.00	Manifest	
Above Disk Plus	65.00	CodeBase 4	195.00	Dynacomm/Windows	279.00	HALO '88	329.00	Map Assist	
Advanced Revelation	595.00	Coherent/PC	125.00	EnerBASE Plus	199.00	Harvard Graphics	279.00	MathCAD	
Allways 1-2-3/Symphony	99.00	Colorix VGA Paint	125.00	Easyflow	149.00	Harvard Project Manager	495.00	MathCAD Add-On's	
Alpha Four	349.00	Command Plus	79.00	Edix	129.00	hDC Fistapps	59.00	Mathematica/386 w/38	
Ami Pro/Windows	325.00	Control Room	79.00	EMACS/DOS	239.00	hDC Windows Express	59.00	Mavis Beacon Typing	
AmiDIAG	75.00	Copy II PC	27.00	EMACS/UNIX	349.00	Headroom	75.00	Media Master Plus	
Animator	289.00	Copywrite w/Zerodisk	53.00	EnerGraphic.	249.00	Hi-Screen/XL	99.00	Memory Mate	
Automate/Anywhere	129.00	Corel Draw/Windows	279.00	Essential Graphics	239.00	Hijaak	89.00	Menugen	
Applause II	279.00	Corporate Ladder	69.00	EXP	269.00	Hooray for Henrietta	19.95	MenuWorks	
Arkanoid II	29.00	Crosstalk Mark IV	109.00	Express Publisher	129.00	Hot Wire	99.00	MetaWindow Plus	
Arts & Letters/Windows	395.00	Crosstalk/Windows	97.00	Extend-a-Name Plus	73.00	Hotshot Graphics	159.00	MicroTSP	
AskSam	185.00	Cruise Control	43.00	Extra K/Symphony	69.00	HyperAccess/5	169.00	Milestones, Etc/Windo	
Asyst Complete	1995.00	Dan Bricklin's Demo II	169.00	FaceLift/Windows	79.00	Hyperpad	99.00	Mirror III	
ATI/Training	59.00	DASM	169.00	Family Tree Maker	49.00	IBM Current/Windows	295.00	MKS Toolkit	
Auto-Might	59.00	Data Junction Advanced	229.00	Fantavision	57.00	IBM Displaywrite IV/V	299.00	Monarch	
AutoImport	129.00	Datease	429.00	Fastback Plus	99.00	IBM DOS 3.x	89.00	Move 'Em	
Automenu	45.00	Datease LAN Pack	569.00	Fastlock	47.00	IBM DOS 4.x	99.00	MS BASIC Compiler	
Autoroute Plus	215.00	DataPerfect	289.00	Fastlynx	85.00	IBM Fixed Disk Organizer	47.00	MS C Compiler	
AutoSketch v3	79.00	dBASE III Plus	327.00	FastText Search/Clipper	79.00	IBM OS/2 Standard Edition	239.00	MS COBOL Compiler	
Avery LabelPro/Lasers	89.00	dBASE IV/Standard	369.00	FastTrax	45.00	IBM Storyboard Plus	229.00	MS Excel/Windows	
B-Tree Filer	69.00	dBFast/Windows	329.00	File Rescue Plus	37.00	IBM Writing Assistant	129.00	MS Flight Simulator	
Baler	310.00	dBrief	79.00	Flowcharting 3	149.00	Illustrator/Windows	289.00	MS F/S Scenery Disks	
Batcom	45.00	dBXL Diamond w/R&R	149.00	Folio Views	379.00	Impress	82.00	MS FORTRAN Compile	
Battery Watch	32.00	Deluxe Paint/Enhanced	79.00	Fontasy 3 Deluxe	79.00	Individual/Training	59.00	MS Macro Assembler	
BeckerTools/Windows	69.00	Derive	125.00	Fontrix	129.00	Info-XL	149.00	MS Mouse + Paint	
Bitstream Fonts (each)	135.00	DesignCAD/2D	159.00	Fontspace	69.00	Info Select	85.00	MS Multiplan	
Blast/PC	159.00	DesignCAD/3D	229.00	Formtool Gold	63.00	Instant Recall	95.00	MS Pascal Compiler	
Blinker	199.00	Designer/Windows	399.00	Formworx/Windows	249.00	Instaplan 5000	449.00	MS Powerpoint/Windo	
Blue MAX	89.00	Desk-Link	95.00	Formworx w/Fill & File	99.00	Intelligent Backup	95.00	MS Project/Windows	
Brainmaker	175.00	DESKpress	229.00	FoxGraph	225.00	Kedit	109.00	MS Quick C	
Brief Editor/DOS or OS/2	179.00	DESQview	79.00	FoxPro	429.00	Kermit for DOS	39.00	MS Quick Pascal	
Brooklyn Bridge	79.00	DESQview/386	99.00	FoxPro/LAN	599.00	KnowledgePRO/Windows	529.00	MS QuickBASIC	
Btrieve for DOS	389.00	dGE Graphics	229.00	Fractools	49.00	Labels Unlimited	42.00	MS Windows SDK	
Builder	99.00	Direct Access	54.00	Framework III	339.00	Language Master	79.00	MS Windows v3	
C-DOC	129.00	Direct/Net	149.00	Freedom of Press	299.00	LANSpool	249.00	MS Word	
C Asynch Manager	149.00	Disk Explorer	59.00	Fresh Utilities	149.00	Lantastic Starter Kit	379.00	MS Word/Windows	
C Tools Plus	109.00	Disk Manager	69.00	Fun School III	21.00	Laplink III	89.00	MS Works	
Cache86	39.00	Disk Manager-N	159.00	Funcky Library	149.00	Laser-Ready	85.00	MSDOS 3.x w/GWBASI	
Calendar Creator Plus	49.00	Disk Technician Advanced	110.00	GB/Executive	169.00	Lattice C	170.00	MSDOS 4.x w/GWBASI	
Canvas/Mac	179.00	Disney Educational	21.00	GEM Artline	389.00	Le Menu	53.00	Multi-Edit Professional	
Captive	30.00	Documentor	179.00	Generic 3D Drafting	329.00	LetterPerfect	139.00	Multi-Lingual Scholar	
Carbon Copy+	99.00	DoDot/Windows	129.00	Generic CADD Level 3	329.00	LimSim	79.00	Multimate 4	
Carbon Copy+ +Host	125.00	DOS Partner	69.00	Genifer III	199.00	LocoScript/PC	125.00	Multimate Advantage II	
Carbon Copy+ AutoPilot	75.00	DosUtils	65.00	Glyphix/WordPerfect	59.00	Lotus 1-2-3 v2.2	269.00	Multiscope/Windows	
Cardbox Plus	225.00	DoubleDisk	79.00	Gofer	59.00	Lotus 1-2-3 v3	339.00	Net.b Library	
Certus/PC	179.00	DR DOS v5	110.00	GoScript Plus	235.00	Lotus 1-2-3/G	650.00	NetThin	
Charisma/Windows	299.00	Dr. Solomon's Toolkit	49.00	Grammatik IV (UK)	95.00	Lotus 1-2-3 for UNIX/386	825.00	Netware C Interfac	
CheckIt	85.00	Dr. Switch	79.00	Grammatik IV/Windows	65.00	Lotus Agenda	275.00	NeuralSoft	
Clarion Pro Developer	549.00	Draflx CAD/Windows	495.00	Grapher	149.00	Lotus Freelance Plus	279.00	New Print Shop	
Clear Plus/C or dBASE	129.00	DrawPerfect	329.00	Graphics Link Plus	139.00	Lotus Magellan	99.00	NewsMaster II	
Clipper v5	325.00	DS Backup Plus	72.00	Grasp	109.00	Lotus Symphony	419.00	Norton Anti-Virus	

THIS IS A SMALL SAMPLE OF OUR RANGE

WE'LL *BEAT* ANY NATIONAL

Software Paradise is an <u>independent</u> software reseller representing more than 2000 manufacturers with over 10,000 software products for IBM Personal Computers (and the Apple Macintosh) running DOS, OS/2 and UNIX. We offer the most comprehensive range of PC software in the world and have been supplying both users and programmers with quality products and the service at the lowest prices since 1986. "The Discerning Person's Guide to Software" is the most complete Buyers's Guide to PC software available and is published quarterly by Software Paradise. Call us on (+44) (0222) 887521 or complete the coupon to receive your **FREE** copy.

FREEPHONE 0800 378·873
ORDER LINE ONLY

PLACE YOUR ORDERS FREE OF CHARGE FROM ANYWHERE IN THE UK

ALL OTHER ENQUIRIES TEL: (+44) (0222) 887521 (10 LINES) FAX: (+44) (0222) 862209 (24 HRS)

OU WON'T FIND A FTWARE ANYWHERE*

for DOS, OS/2, Mac & UNIX

AL ● EXPERT SYSTEMS & A.I ● GRAPHICS ● HARD DISK BACKUP ● INTEGRATED SOFTWARE ● LANGUAGES ● LINKERS ● MAC SOFTWARE ● MATHS & STATS ● MEMORY MANAGEMENT ● MISCELLANEOUS ● NETWORK SOFTWARE ● OCR & SCANNING ● OPERATING SYSTEMS ● PIM'S & HYPERTEXT ● PRINT SOFTWARE

129.00	Norton Backup	69.00	Pizazz Plus	79.00	RTLINK Plus	329.00	Toolbook/Windows
125.00	Norton Commander	79.00	PKZIP/PKUNZIP	54.00	Saber Menu/DOS	89.00	Top Priority
39.00	Norton Editor	49.00	PlanPerfect	235.00	Saber Menu/Windows	449.00	Touch & Go
229.00	Norton Guides C, C++ etc	57.00	Playroom	37.00	SCO XENIX/386 Complete	1275.00	Tree Diagrammer
345.00	Norton Utilities v6	109.00	Plink-86 Plus	349.00	Security Guardian	229.00	Turbo Asynch Plus
75.00	NoSquint	45.00	PolyBoost II	65.00	Seagate 1-2-3	65.00	Turbo C++
595.00	Nutshell Plus	189.00	PolyMake	110.00	Service Diagnostics Kit	395.00	Turbo C++ Professional
35.00	ObjectVision	299.00	PolyShell	79.00	Show Partner F/X	245.00	Turbo Pascal
55.00	Object Professional	99.00	Popdrop Plus	75.00	ShowText	149.00	Turbo Pascal Professional
45.00	OmniPage-386/Windows	529.00	Portex Professional	117.00	Sidekick Plus	97.00	Turbo Professional
45.00	Open Access III	479.00	Power BASIC	79.00	Sidetalk	79.00	TurboCAD
35.00	Opt-Tech SORT	125.00	PowerMenu	69.00	Sidaways	39.00	Twist & Shout
185.00	Option Board Deluxe	105.00	Powerpak/Windows	69.00	Sigma Plot	445.00	Type Manager/Windows
499.00	Optune	59.00	PreScript Deluxe	295.00	Silvercomm Library	180.00	Typing Tutor IV Plus
129.00	ORCAD/PCB	1395.00	Presentation Task Force	249.00	Sim City	26.08	Typografica Prime
169.00	ORCAD/SDT	495.00	Primetime Personal	75.00	SitBack	62.00	UI Programmer v2
169.00	Org Plus Advanced	85.00	PrintAPlot	165.00	SiteLock w/Anti-Virus	329.00	UI Touch & Go
299.00	Overlay() for Clipper	99.00	Printcache	95.00	Smalltalk V/286	129.00	UltraScript
57.00	P.D.Queue for 1-2-3	59.00	Printer Assist	149.00	Smalltalk V/PM	295.00	UltraScript PC Plus
239.00	PageGarden	63.00	Printmaster Plus	49.00	Smarterm 240	179.00	Ultravision
229.00	Pagemaker/Mac	569.00	PrintQ	89.00	Smart System	469.00	Vcache
489.00	Pagemaker/Windows	453.00	Printrite	85.00	Smartware II	469.00	Ventura GOLD/Windows
259.00	Paradox v3.5	379.00	Procomm Plus	72.00	Soft Kicker Plus	95.00	Vermont Views
39.00	Paradox Engine	269.00	Procomm Plus w/Viewdata	107.00	Soft Type/Windows	129.00	Virex-PC or Mac
19.00	Paradox SQL Link	269.00	Professional File	169.00	Softbytes	59.00	Virus Hunter
229.00	Pathminder Plus	79.00	Professional Write	229.00	Software Bridge	85.00	ViruSafe
85.00	PC Anywhere III/IV	110.00	Professor DOS	57.00	Software Carousel	57.00	VM/386
92.00	PC Guide/Windows	429.00	Project Calc/1-2-3	139.00	Sourcer/486 w/BIOS	125.00	VolksWriter 4
109.00	PC Paintbrush IV Plus	159.00	Project Scheduler/Windows	449.00	SourcePrint	69.00	Vopt
159.00	PC Tools Deluxe	79.00	Project Workbench	795.00	SPF/PC	169.00	Watchdog
295.00	PC Write	85.00	Prokey Plus	79.00	Spinrite II	89.00	Where in Europe
349.00	PC-Browse	55.00	PVCS Professional	449.00	Sprint	135.00	Who-What-When
59.00	PC-File 5	79.00	PVCS/LAN	1595.00	SPSS/PC+	99.00	Windows Workstation
57.00	PC-Fullbak Plus	69.00	Q&A	239.00	StatGraphics	429.00	Wingz/Windows
59.00	PC-Globe	69.00	QDOS-II	49.00	Sterling Accountant Plus	329.00	Wonder Plus
275.00	PC-Guard/D Plus	137.00	QEdit TSR	79.00	Sterling Financial Controller	429.00	Word for Word Pro
75.00	PC-Kwik Power Disk	59.00	QEMM/386 w/Manifest	57.00	Stonybrook Modula-2	229.00	WordPerfect
253.00	PC-Kwik Power Pack	83.00	Qram w/Manifest	57.00	Storyboard Plus	229.00	WordPerfect Addn'l Users
299.00	PC-Lint	89.00	Quattro Pro	269.00	Super PC-Kwik	53.00	WordPerfect Office
99.00	PDC PROLOG for DOS	199.00	Quick Schedule Plus	59.00	Superbase 4/Windows	449.00	WordPerfect/UNIX
79.00	Pegasus S/U Modules	205.00	QuickPak Professional	129.00	Supercalc 5	75.00	WordScan Plus/Windows
89.00	Pegasus Senior Modules	429.00	R&R Report Writer	135.00	SuperFonts	125.00	Wordstar 2000+
139.00	PerFORM Pro/Windows	269.00	R&R Clipper/Fox Module	49.00	Superkey	55.00	Wordstar Pro
325.00	PerFORM/GEM	159.00	R&R Code Generator	99.00	SuperPrint/Windows	149.00	XENON II MagaBlast
299.00	Personal COBOL	129.00	R:Base for DOS	379.00	SuperProject Expert	459.00	XtreeNET/ELS
289.00	Personal Measure	69.00	Rapidfile	195.00	Surfer	339.00	XtreeNET/Advanced
329.00	Personal REXX	129.00	Reader Rabbit	35.00	Switch-It	72.00	Xtree Pro
149.00	Perspective Junior	129.00	ReadRight/Windows	429.00	Systat w/Graphics	599.00	Xtree Pro Gold
79.00	Pertmaster Advance	1639.00	Reflection 1+	199.00	System Sleuth	85.00	XYwrite III Plus
185.00	PFS:First Choice	123.00	Reflection 4+	229.00	System WinSleuth/Windows	85.00	XYWrite III Plus/LAN
139.00	PFS:First Graphics	117.00	Reflex v2	179.00	Take II Manager	79.00	ZINC Library
46.00	PFS First Publisher	125.00	Relay Gold	179.00	Talking Windows	235.00	Zortech C++ Video Course
59.00	Picture Publisher	449.00	Remote 2 Complete	129.00	Timehne v4	479.00	Zortech C++ Developer
129.00	Pixie/Windows	169.00	RM/FORTRAN	399.00	Timeworks	85.00	

CALL US FOR ANY ITEMS NOT LISTED

LY ADVERTISED PRICE+

OUR POLICY

- All prices and manufacturer's specifications are subject to change without notice.
- All prices exclude carriage and VAT.
- All orders subject to availability
- Please specify media size when ordering.
- Goods are not offered on a trial basis.
- Goods are offered for sale subject to our standard conditions of sale which are available on request. E&OE.

CIRCLE NO. 497

VISA

* Provide written proof of any U.K. mail-order software company who supplies a wider range of IBM PC software than we offer in our catalogue within 14 days of this advertisement and get yourself a FREE copy of PC Tools Deluxe!

+ We are always delighted to be given the opportunity to beat other nationally advertised prices (at the time of placing an order) subject to the same terms and conditions of the company being beaten.

All Trademarks and Registered Trademarks are hereby acknowledged.

SOFTWARE PARADISE

The Software Experts

Software Paradise, Avenue House, King Edward Avenue, Caerphilly, Mid Glamorgan, CF8 1HE U.K.

FREEPOST — TODAY —

NO STAMP NEEDED

(U.K. ONLY)

☐ Please send me my *FREE* 32 page copy of *'The Discerning Person's Guide to Software'*

NAME

COMPANY

ADDRESS

POSTCODE **TEL**

Post now to: Software Paradise, FREEPOST, Avenue House King Edward Avenue, Caerphilly, CF8 1ZZ

EXE 4/91

CIRCLE NO. 690

G.W. Computers Inc.
ADCI, 4 Eagle Square,
E.Boston. MA02128
U.S.A.
Tel: 617-569-5990
Fax: 617-567-2981

Xvale Ltd., Store D
55, Bedford Court Mansions,
Bedford Avenue, London
WC1B 3AD U.K.
Tel: 071-636-8210
Fax: 071-255-1038

We sell Hundreds of brand name DOS and UNIX software, and offer wide in depth experience on networks. CALL US!!.

LINX SYSTEMS
STANDARD FEATURES

1 * 5-1/4 " drive
1 * 3-1/2 " drive
Dual HD/FD Controller
101 Key Keyboard
Mini - tower case
2 Serial/1 Parallel/1Game
Built in Diag / AMI bios
Mono Screen / Adapter

CPU/ Mhz+Ram = Price

8088 /10+640k	£545
80286/12+2mb	£745
80286/16+2mb	£825
80386/16+4mb	£1145
80386/20+4mb	£1345
80386/25+4mb	£1495
80386/33+4mb	£1745
80486/25+8mb	£2695

**** For SVGA Colour****
Add Screen/Card £245.00

Then just add the
HARD-DISK of your choice

** LAPTOP portable **
80286-12,vga,40mb £2150.00
** NOTEBOOK portable **
80386-16,vga,20mb £1950.00

Unix 4 port serial £145
Unix 8 port serial £195
Intelligent 4 serial £395
Genius mouse 3but £45

Hard-Disks

20 --	£150.00
40 --	£195.00
80 --	£325.00
100 -	£495.00
120 -	£595.00
200 -	£755.00
300-	£1095.00
600-	£1495.00

Ethernet

wd8bit	£135
wd16bit	£155
lnx8bit	£110
lnx16bit	£135

INTERACTIVE R 2.2 UNIX PLATFORM

Application	2 usr	£495
Application	3+usr	£1095
Network	2 usr	£695
Network	3+usr	£1350
Workstation	2 usr	£855
Workstation	3+usr	£1495

UNIX DEVELOPER

Application	2usr	£1095
Application	3+usr	£1750
Network	2usr	£1155
Network	3+usr	£1815
Workstation	2usr	£1295
Workstation	3+usr	£1945

SCO

Xenix o/sys	£422
Xenix s/dev	£455
Xenix tx/prc	£126
Xenix cmplt	£905
Xenix tcp/ip	£227
Xenix net	£451
Unix o/sys	£525
Unix s/dev	£613
Unix cgi	£191
Unix m/vw	£321
Unix vpix	£321
Unix tcp/ip	£227
Unix nfs	£340
Open dsktp	£795
O/dtp s/dev	£1095
O/dtp servr	£1095
Word 5.0	£650

HOT ITEMS

Guru reprter	£49.95
Syseditor	£49.95
Cshell	£49.95
Menuix	£49.95
Sysix	£99.00
Sysix Pro	£145.00
SystemIX	£295.00
PC-Connect	£450.00
Pc-interface	£170.00
PC-TCP	£375.00
PC-Xview	£245.00
Terminl ctl	£695.00
Xvision	£295.00
Coherent ux	£99.00
DosBridge	£295.00

*** BUNDLED SYSTEMS include software Value up to £1730 FREE ***
Buy a 386/33 mhz with SVGA screen and 118 mbyte h-disk
at £2535.00 , get * INTERACTIVE UNIX Sys V, Rel 3.2, V2.02 *
RUNTIME / VPIX and SOFTWARE DEVELOPMENT + MENUIX
OR RUNTIME / TCP-IP and XWINDOWS + MENUIX
FREE !, CALL for latest deals. (24hr FAX)

** UNBEATABLE SYSTEM BUNDLES includes *FREE SOFTWARE* **

LINX 8088 - 10	LINX 80286 - 12	LINX 80386sx-16	LINX 80386-33	LINX 80486-25
mono/20mb	mono/40mb	colour/80mb	colour/120mb	colour/200mb
£695.00	£1020.00	£1760.00	£2535.00	£3695.00
£270.00 FREE	*£370.00 FREE*	*£800.00 FREE*	*£1300.00 FREE*	*£1730.00*

CALL FOR DETAILED SPECIFICATIONS
and any special offers

Limited time offer on any item of software IN HOUSE while stocks last !!. Basic deals are intended to be 386 system UNIX SYSV, REL3.2,Ver2.02 Workstations CALL NOW and reserve yours !!

CIRCLE NO. 691

National characters drives us mad

Denmark's Ebbe Sønderby explains the difficulties of using computers in a country where the language requires an extended alphabet.

Danish is not a language spoken or read by many people in the world. This fact is exploited by some of my coarser-natured compatriots, who feel free to comment frankly on anyone and anything they experience when travelling abroad.

Apart from the fact that the Danish language is spoken cryptography, the language is blessed with - or cursed with - three letters not normally found in English, namely 'Æ', 'Ø' and 'Å'. Æ is not all that unusual, and Å is also used in Sweden and Norway. Ø, on the other hand, to my knowledge is only used in Denmark and Norway, although it appears in the form ö in Sweden and Germany.

The Danish alphabet ends with the letters Æ, Ø and Å in that sequence. For decades, these letters were no problem; but then the computers came.

Computer standards are more or less *de facto* standards originating from English speaking countries. Though Denmark has a computer industry of its own, it certainly had no influence on ASCII. Æ, Ø and Å were completely missing, so how to use standard equipment in Denmark?

The answer was a simple one in the happy ASCII days: Z was the last letter in ASCII, followed only by the character sequence [\] or {|} for upper and lower case letters respectively. At that time [\] and {|} were not used for much in Denmark except in mathematical text books, and such books were the territory of specialist book printers and had nothing to do with computers. (The { and } are known locally as the *Tuborg parenthesis*, after the famous Danish brewery which uses a curly bracket shaped umbrella on top of its trucks. The [\] and {|} sequences were replaced by **ÆØÅ** and **æøå** in the character generators on CRTs and printers. New keytops were made. This was irritating but not difficult and everything worked out well. Eventually even the international hardware manufacturers noticed the problem, and installed DIP switches to remedy it.

Danish programmers did not have much difficulty. Since programming languages do not use the special Danish characters, software was developed on unmodified equipment which had original character generators and original English keyboards. Comments - if any - were normally written in English. Program descriptions and other documentation likewise. Only special user manuals were written in Danish, and they were typed by secretaries - on typewriters. Things were all right for a time.

The curse of PCs

Then the PCs emerged. There were Apple IIs and the other products of that time. Schoolboys learned to accept, that if you wanted a Ø, you had to type \.

Then came the IBM PC. IBM claims that the 'I' in IBM stands for 'international'. Danes know better. The correct word is another one starting with 'I' and ending with 'compatible'. Though the Æ, Ø and Å were baked into at least one of IBM's character sets, there were problems, especially with the ø.

First of all, the characters could not print on the old, modified printers. This did not matter much; buy a new IBM compatible and you were all right, except for the fact that Ø would sometimes show like Ø on the monitor and print as ¥ and ø as would show as ø and print as ¢, or the other way around. This was caused by the fact that there are two, rather than one, IBM character sets. Strange AUTOEXEC.BATs and CONFIG.SYSs including code page commands would sometimes cure this problem, but every time a new peripheral, was connected new problems would appear. Sometimes the peripheral wanted the old ASCII variant, sometimes it had the wrong IBM character set. Some packages supported ø, but didn't always print it and so on.

Then there is sorting. The sequence of Æ, Ø and Å in the Danish alphabet is not maintained in the IBM character set; normally it will sort in the sequence Å, Æ, Ø. But not always. If, for example, code pages are not used because the VGA card then shows the wrong characters, the sorting sequence in WordPerfect will depend on the code page chosen as a start up parameter. There is a tiresome and constant programming problem here; its solution is not entirely trivial.

Ø is not the only problem the Danes have. One thousand and a half is written '1,000.5' in many countries, but in Denmark it is '1.000,5'. It may seem like a small thing, but often a Dane will use an English version of a program to perform part of a task, then change to a Danish version of another program to finish it off. This means that a mental change from point to comma must be made, an almost impossible feat.

It is hard to write programs for export to other countries, and be sure that every requirement of that country is fulfilled. It is not hard to make the software flexible enough if the requirements are known, but sometimes the problems can catch you out. In this article, I have described the pitfalls in Denmark. Because the Danes are so painfully aware of the differences between Danish system requirements and other requirements, we take pride in the fact that *our* stuff is designed to be compatible with every market. I believed in this piece of received wisdom myself, until one day I had to deliver a Danish system to Portugal. The equipment included currency displays and counters for petrol tanks. But nobody had remembered that, in Portugal, currency displays and total counters to have lots of digits!

Some day, I am sure, all these problems will be over. But perhaps everyone will speak English by then...

EXE

Ebbe Sønderby graduated with an MSc in Electronic Engineering from the Technical University of Denmark in 1976. He now works as a project manager at a Danish Asea Brown Boveri subsidiary. His mission in life is to make products combining hardware and software in the best possible way.

Nil desperandum sampling...

is dog Latin for 'simulated random sampling is not to be despaired of.' Robert Loughrey takes the Code Page to a higher plane, and finds that statistics can be useful.

The final stages of data acquisition programs can be a problem. Vast amounts of dummy run test data must be generated to reassure programmer and client alike that all is well with the software, and maybe - especially if the package is an off-the-peg one - a demonstration program must be cobbled together with realistic simulated input.

For instance: external hardware - maybe an ADC reading 'real' voltages - is connected to a port of your computer. Your program module samples this port periodically, and fits the closest possible normal distribution to it (where 'closest possible' has some meaningful definition). Certainly, your client will now want to see a stream of artificial test data generated in software with known parameters to see how close your module gets, before it is integrated into the complete system. At the same time, Sales wants you to write a demo program that is interactive, and hence must 'create' fresh realistic-looking data each time it is run. They don't want to drag the ADC about with them when they go to see the clients, so again it must be done completely in software. How can these data streams be provided?

As a more camp example, your latest executive computer game simulates the stock market. You already know the *mean* of any given share or stock in any game round (ie, from its value in the previous round and from other variables representing 'rumour', 'confidence', 'annual profits', etc). Maybe you also know how bullish or bearish you want the market to be - its *standard deviation*. How can you use these statistics to generate a specific value for the commodity? The value must be *plausible*, so it will probably be somewhere around the mean (but might not be) even though it could conceivably be any value (but isn't likely to be, especially in a bearish market).

rand() has been shown to be not quite as satisfactory as was thought in K&R's heyday

These examples belong to essentially the same class of problem. One solution is to measure the heights of everyone in the office, correct the data for sex, age and race, store the distribution as a table in the program, and 'shuffle' that table before use so that it isn't, for instance, simply in increasing order. This solution is somewhat unsatisfactory. Fortunately, another solution exists: statistical simulated sampling.

The Normal distribution

It is a classical, almost philosophical, problem in science from Quantum Mechanics to Genetics to predict the behaviour of individuals from given knowledge of collectives. Usually, no certain solution exists (or is economic to find). But propositions about the probability of specific individuals behaving a certain way can be deduced from knowledge of patterns of behaviour.

If you were asked to find the pattern that your port samples or shares were to form as a whole, the answer, in all likelihood, would be something like:-

((Equation 1)) $$\frac{1}{\sqrt{2\pi}s} e^{-(\frac{1}{2})(\frac{x-m}{s})^2}$$

(usually abbreviated to $N(m,s^2)$) where m is the mean and s is the standard deviation. This is the (density function of the) celebrated *Normal Distribution* (Figure 1, where $m = 0$ and $s = 1$, should be familiar to the most 'mathophobic' reader) which captures the essence of what we mean by 'Random Behaviour'; and, indeed, thanks to a result called the Central Limit Theorem, empirical distributions from a wide range of situations tend to conform closely to the

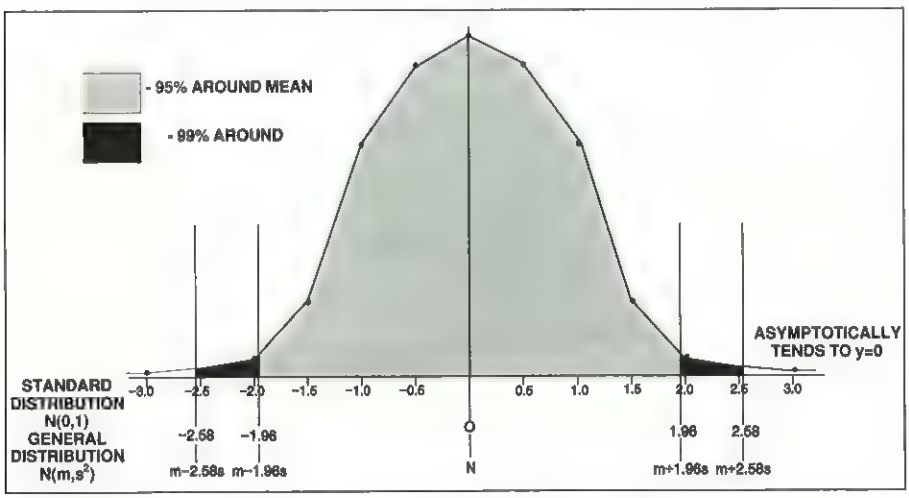


Figure 1 - The Normal distribution

above formula. So, the sizes of your friends feet, the proportion of 'heads' achieved by tossing a fair coin, and the IQs of children in a school all fit into a pattern that looks like Figure 1.

Provided, that is, the sample is large enough; you must start with a large number of friends, tosses, or children. In particular, $N(m,s^2)$ says nothing about any specific instance of a random variable. For example: you suspect, and your software is designed to assume, that your port reading follows a normal distribution with mean 0 and standard deviation 1: $N(0,1)$. Well, you can read immediately from Figure 1 that 95% of all readings will be between ±1.96. So, is the next reading from the port going to be within that range?

And did those feet

Predicting with certainty any particular value of a 'random' variable like this is, of course, no more possible then that I could correctly guess the size of your feet. But from a knowledge of your sex, age, and race, I can deduce the probability that you take a particular shoe size. For, the probability that a random variable falls within a range is given by integrating the density function of the variable between the limits of the range.

We know foot size closely fits a Normal Distribution (it is the sum of a great many Binomial - genetic - variables, and hence is Normal by the Central Limit Theorem). Suppose that the results of measuring the feet of a great many other people of your sex and age enables us to say that the mean of this distribution is 25cm and its standard deviation is 2cm. For the sake of argument, everyone with a *foot* size between 23 and 27cm takes a *shoe* size 8. Then the probability of you taking size 8 is the definite integral of $N(25,2^2) = N(25,4)$ between the limits 23 and 27, or .6826. The probability of you taking any shoe at all (including impossible ones like -3.142) is the definite integral of $N(25,4)$ between the limits -∞ and +∞; realist readers will be pleased to hear that this evaluates to 1, and to know that it will do so independent of the initial choices of m and s.

Simpson Integration

Unfortunately, it is not usually at all easy - or even possible - to evaluate these kind of integrals symbolically. Generally, they are calculated numerically. Of course, for the Normal Distribution, tables of definite integrals of the Normal density function are fairly easily obtainable; typically, these give the integral between -∞ and x for $N(0,1)$. A few values from these tables are given in Figure 7. For instance, integrating $N(25,4)$ between the limits 23 and 27 is the same as integrating $N(0,1)$ between (23-25)/2 = -1 and (27-25)/2 = +1. From the symmetry of the density function (see Figure 1), this is twice the integral of $N(0,1)$ between -∞ and 1, minus 1; or

(2 x 0.8413) -1

= .6826 to 4dps, as before (see Figure 7).

It's not really practical to store tables in a program, so your software must itself numerically integrate the density function. It is fastest to calculate any given definite integral of $N(m,s^2)$ directly; for instance, from a power series. In this article I shall instead use Simpson's Method, partly because your statistics may fit a non-normal distribution, but also because Simpson's method provides a slick, general purpose numerical integrator; something that, arguably, should be in everyone's maths library anyhow.

```
double simp(a, b, n, funct)
double a,b;
int n;
double (*funct)();
/* Integrate funct between a and b,
   using n ordinates.
   To use this code, call with (for instance)
         double x = 10.0, y;
         y = simp(0.0, x, 8, poly);
   note:
     header given in pre-ANSI form for clarity.
     Calling procedure must ensure that n is
     even & positive and that b is greater
     than a.                                    */
{
  int count;
  double sum, value, interval;
  sum = 0.0;
  value = a;
  interval = (b-a)/n;
  for(count=1; count<n; count++)
  {
    value += interval;
    if (count%2 == 0)
      sum += (*funct)(value);
    else
      sum += 2 * (*funct)(value);
  }
  sum *= 2.0;
  sum += (*funct)(a) + (*funct)(b);
  return(interval * sum / 3.0);
}

double poly(double x)
/* demonstration polynomial function */
{
  return( (3*x*x) + (2*x) + 1 );
}
```

Figure 2 - C source for Simpson integration

Simpson's Rule - after Thomas Simpson (1710-1761) - is widely acknowledged to be the best all round method of *Quadrature*, blending speed with accuracy and ease of programming. My listing for it (Figure 2) takes a pointer to a (C language) function as a parameter and is thus perfectly general; so, stick it on your utility disk and forget about it until you next need to calculate the orbit of Saturn or whatever.

Figure 2 is fairly self-explanatory, once you know that Simpson's Rule says that an ap-

Simpson of Simpson's rule

Thomas Simpson - 1710-1761 - isn't really famous for many things. Born in Leicestershire, he was a weaver by trade. In 1730 he married a woman twenty years his senior, and who would survive him, living to be 102. In 1733 he was banished from Nuneaton, where they lived, for practising witchcraft - actually a common hobby amongst men of science in those days. In London, he slowly rose to scientific respectability, and was elected fellow of the Royal Society in 1745. Between 1754 and 1760 he was editor of *The Ladies' Diary*, a kind of Georgian *Bella*. The method of quadrature that bears his name is not his only mathematical work by any means, and his general output includes tracts on calculus, geometry, experimental method and algebra. His technique was always a productive mixture of the plodding and the ingenious; yet somehow, he never again broke through from the interesting into the truly original.

```
#include <math.h>
#define PI 3.142

double normal(double mu, double sigma, double x)
/* returns the density function of the Normal curve
     N(mu, sigma-squared) at x                        */
{
/* first should really ensure sigma > 0. Then return
          exp( (mu-x)*(mu-x) / (2.0*sigma*sigma) )
          ----------------------------------------
               sigma * sqrt(2.0 * PI)              */

  double dividend, divisor;
  dividend = exp( (mu-x)*(mu-x) / (2.0*sigma*sigma) );
  divisor = sigma * sqrt(2.0 * PI);
  return dividend/divisor;
}

double stand_norm(double x)
/* returns N(0,1) */
{
  return( normal(0.0, 1.0, x) );
}

double phi(double z)
/* returns probability that the value of a
   random variable distributed N(0,1) is <= z? */
{
  if (z>0)
    return( 0.5 + simp(0.0, z, 8, stand_norm) );
  else
    return( 0.5 - simp(0.0, -z, 8, stand_norm) );
}

int scale(int n, double mu, double sigma, double x)
/* scales normal() for drawing on screen with n rows */
{
  return( (int)(sqrt(2.0*PI)*sigma*n*normal(mu,sigma,x) )
);
}

double sample(int mu, int sigma)
/* simulate a sample from normal curve
   with mean mu and sd sigma.
   Pass sigma=0 to use previously
   calculated allocations. */
{
  static int allocated[1001];
  int low, high, count1, count2, k;
  if(sigma)
  {
    low = mu - 3*sigma;
    high = mu + 3*sigma;
    for(count1=low, count2=0; count1<=high; count1++)
    {
      k = (int)(1000.0 *
        phi((double)(count1-mu)/(double)sigma));
      while(count2 < k)
      {
           allocated[count2] = count1;
           count2++;
      }
    }
    while(count2 < 1000)
    {
         allocated[count2] = count1;
         count2++;
    }
  }
  return( allocated[nrand(1000)] );
}
```

Figure 3 - C source of the algorithm

proximation to the definite integral of $f(x)$ between the limits a and b is given by:-

((Equation 2)) $$\frac{h}{3}(y_0+4y_1+2y_2+4y_3+\ldots+2y_{n-2}+4y_{n-1}+y_n)$$

The Interval $[a,b]$ has been split into n equal subintervals of width h; yr is the value of $f(x)$ at the right end ('ordinate') of the r^{th} subinterval (ie, when $x = a + rh$). It follows that $nh = b-a$.

In the listing for the function `simp()` I have passed n as a parameter. You can set this to any (positive even) integer, but values greater then 8 are starting to give more accuracy than is required, and may slow the routine down unacceptably. In addition, although increasing n obviously decreases the *discretisation* or *truncation* error - that is, the theoretical error due to the fact that, in deriving any method of quadrature, $f(x)$ is approximated by a simpler polynomial - it increases the *round-off* error - the practical problem of finite computer word size. In general, the discretisation error of Simpson's Rule will be proportional to h^4, while the upper bound on the round-off error is inversely proportional to h; so the latter rapidly catches up with and overtakes the former.

You might like to try `simp()` on the demonstration function `poly()` (integrating `poly()` between 0.0 and 10.0 ought to give you 1110.0), your own functions, or even standard calls to the maths library like `sin()` or `exp()`. Or try `simp()` on the normal equation above. Confirm that integrating $N(25,4)$ between 23 and 27, for instance, gives .6826.

nrand() and snrand()

These two functions, shown in Figure 4, replace the table of random numbers in a manual calculation, and have been lifted more or less wholesale from K&R's listing (p46 in the Second Edition). They rely on the *Linear Congruential* method - linearly transforming the last number in the sequence and dividing it modulo the upper limit of the random range to get the next. `nrand()` takes one integer parameter and returns a pseudo-random non-negative number less then that parameter. `snrand()` 'seeds' this function - it provides an initial entry point. Ideally, the parameter passed to `snrand()` should itself be a random number, although in practice using the seconds in the current minute or the ticks used in the current time-slice are both forgivable skimps.

Otherwise, the functions are used in the same way as `rand()` and `srand()` in the standard maths library. It cannot be emphasised enough that the values returned by `nrand()` are only *pseudo-random* - in fact, deterministic, but with behaviour sufficiently complex to appear random. In recent research, linear methods like those used in `rand()` have been shown to be not quite as satisfactory as was thought, perhaps, in K&R's heyday. In particular, `rand()` has an 'effective' period of only 16000, much less then was previously thought.

Lies and Stats

Figure 3 ties together all the work done so far. The function `sample()` returns a normally distributed (integer, for simplicity) value on $N(m,s^2)$, where m and s are passed

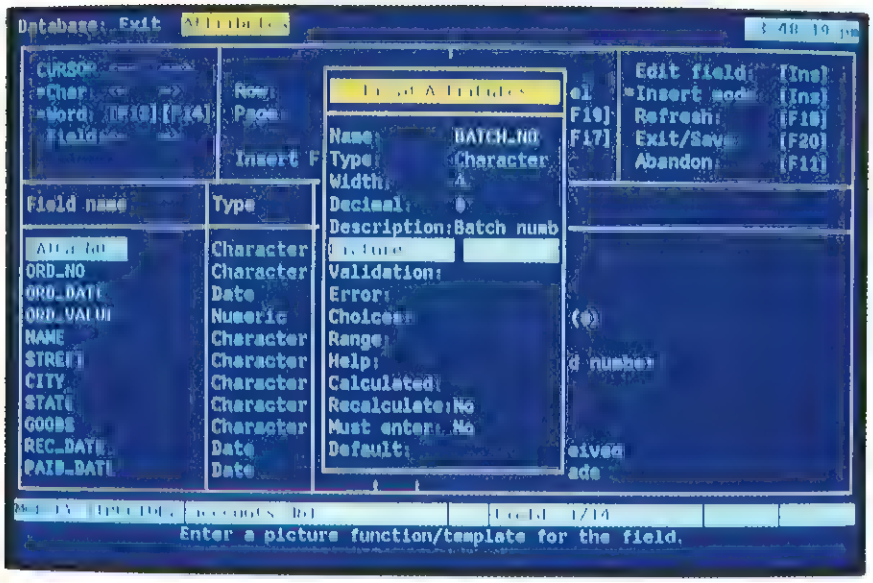
```
long int next = 1;

int nrand(unsigned int n)
/* Returns pseudo-random integer on 0,..., n-1 */
/* n >= 2 */
{
  next = next * 1103515245 + 12345;
  return( (unsigned int)(next/65536) % n);
}

void snrand(unsigned int seed)
/*  Set seed for nrand() */
{
  next = seed;
}
```

Figure 4 - C source for nrand() and snrand()

dBase III Plus, FoxBASE & Clipper applications running on VAX & UNIX?



Try our 30-day software licence
In an hour or two you'll be converted

Can Recital really be that easy?

The answer is yes. And to prove it, we invite you to send for our full-featured 30-day software licence for VAX/VMS™ and UNIX systems. Recital is so compatible and easy to use, you'll have your dBase™ applications up and running in Recital in about an hour. Without any additional programming or modifications.

Then take the rest of the month to discover the richness and power of this 4GL system. Recital includes tools like a report writer, screen painter, transparent access to RMS files, an integral data dictionary, popup calendars, calculators, pick lists, note pads and much more. All on a VT-type terminal!

Your 30-day license will include the full Recital system, documentation and hotline support. How can you lose?

Order today by calling one of the numbers below. Or write to Recital Corporation.

Recital Corporation Limited
South Bank Technopark
90 London Road
London SE1 6LN, UK
Tel 071-401 2727
Fax 071-633 9617

Recital
CORPORATION

Recital Corporation Inc
85 Constitution Lane
Danvers
MA 01923, USA
Tel (508) 750 1066
Fax (508) 750 8097

All trademarks are the properties of their respective companies

CIRCLE NO. 692

value	standardised	integrated	allocation
19cm	-3.0	.0013	0 to 13
20cm	-2.5	.0062	14 to 62
21cm	-2.0	.0228	63 to 228
22cm	-1.5	.0668	229 to 668
23cm	-1.0	.1587	669 to 1587
24cm	-0.5	.3085	1588 to 3085
25cm	0	.5	3086 to 5000
26cm	0.5	.6915	5001 to 6915
27cm	1.0	.8413	6916 to 8413
28cm	1.5	.9332	8414 to 9332
29cm	2.0	.9772	9333 to 9772
30cm	2.5	.9938	9773 to 9938
31cm	3.0	.9987	9939 to 9987
32+cm			9988 to 9999

So, for instance, generating the random number 7345 would correspond to foot size 27cm.

Figure 5 - An example using the Normal distribution

Note: since it is not practical to integrate xI, we instead merely sum values of the density function. For instance, .0183 + .0733 + .1465 = .2381.

value	density	funct	'integration' allocation
0	.0183	.0183	0 to 183
1	.0733	.0916	184 to 916
2	.1465	.2381	917 to 2381
3	.1954	.4335	2382 to 4335
4	.1953	.6288	4336 to 6288
5	.1563	.7851	6289 to 7851 etc.

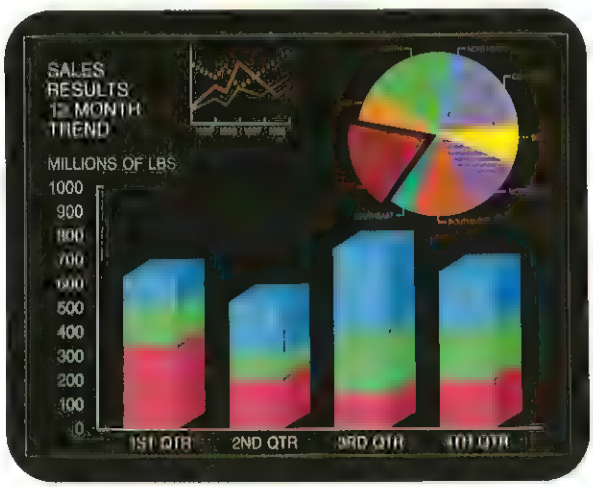
Figure 6 - An example (non-Normal distribution)

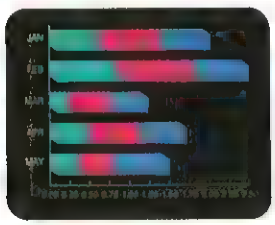
parameters. To see how it works, recall that we can deduce from Figure 7 that practically all (99.74%) of the distribution is covered by the range ±3s. We next allocate groups to each value within the range; the group number increases with increasing x, and the number of groups allocated to any value is proportional the (standardised) probability of that value being the closest to a random sample.

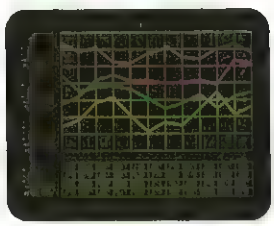
For example, let's follow the process with a particular distribution: feet sizes (Figure 5). As before, feet for a particular age and sex are normally distributed with mean m = 25cm and standard deviation s =2cm. So 99.75% ('all') of measured samples fall in the range x_i = 19 to 31cm. Standardise each of these values - map from $N(25,4)$ onto N(0,1) - by replacing with $y_i = (x_i - m)/s$ = -3, -2.5, -2, ..., 2.5, 3. Next integrate N(0,1) from $-\infty$ to y_i, giving .0013, .0062, ..., .9938, .9987. These form the basis of our allocation, 0 to 13, 14 to 62, etc.

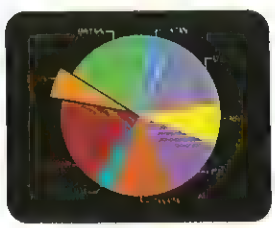
For speed all the above is done only once for each distribution, then stored in the static array `allocated[]` (pass `sigma != 0` for `sample()` to re-evaluate the array). So, for instance, in the above

dGE
VERSION 4
GRAPHICS FOR dBASE









dGE extends the database language of your choice to give full colour graphs and charts from **within** your programs

dGE version 3 was voted the "best graphics tool for the database manager" for the second year running by readers of the influential U.S. magazine DataBased Advisor.

Now Version 4 is here with even more features.

- SuperText with variable sizes and face styles
- SuperIcons loaded from file
- SuperText and SuperIcon editor that can import PCX images
- Built-in Vector Text functions
- Gantt charts
- XOR capability added to many functions
- New interfaces to Clipper 5, FORCE and Vulcan
- Mouse "hot" regions
- A Norton Guide to functions

dGX, the Graphics Design Centre, has been completely revised, adding.

- Sophisticated database query and summary
- Dynamic scaling
- Import of PCX images
- Statistics functions
- Gantt charts, hi-lo and 3-D bars

Version 4 comes complete with all the interfaces to dBASE and its dialects, the 'C' language, R:BASE compiler and Vulcan.

dGE supports CGA, EGA, VGA and Hercules modes and will output images to matrix printers, lasers and plotter.

Call for information.

Price £245 + VAT direct from Bits Per Second Ltd or from your authorised dealer.

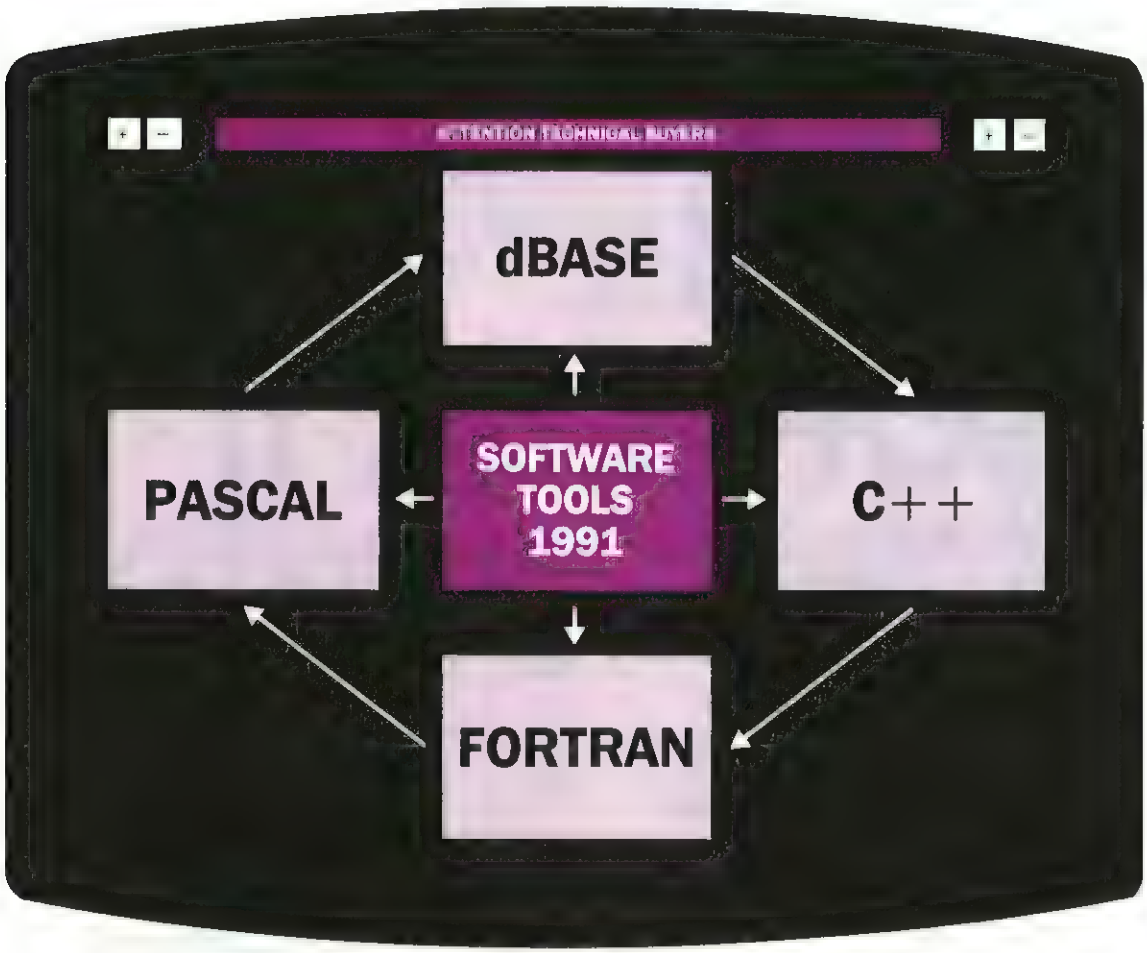
dGE IS A TRADEMARK OF BITS PER SECOND LTD. OTHER BRAND AND PRODUCT NAMES ARE TRADEMARKS OF THEIR RESPECTIVE HOLDERS

BITS PER SECOND

BITS PER SECOND LIMITED
14 REGENT HILL BRIGHTON BN1 3ED
TELEPHONE: 0273 727119
FAX: 0273 731925

CIRCLE NO. 693

Meet people who speak your language



Whatever your specialist interest you'll meet someone at Software Tools '91 who shares it. Because at Europe's largest and best event for software development you'll find more than just sales talk.

You'll meet people who really understand their products AND your requirements. And you'll have the chance to roll up your sleeves and get some real 'hands on' experience on the latest products. All in all, the perfect environment to find real solutions.

Special features at this year's show include:

- A PC Development Area
- A People and Skills Centre
- A Technical Preview Centre
- An exciting line-up of competitions and awards

All this, and a conference which includes some of the best known experts in the field talking about all that's current – and controversial – in software development today.

For more info plus your free tickets to Software Tools '91 just return this coupon to Lynne Davey, Blenheim Online, Blenheim House, Ash Hill Drive, Pinner Middx HA5 2AE, before 3rd May 1991.

11-13 June 1991 Wembley Exhibition Centre, London.

Sponsored by EXE

Software MANAGEMENT

Supported by

Name _____

Initials _____ Mr/Mrs _____

Company _____

Job Title _____

Address _____

_____

_____

Tel _____ Fax _____

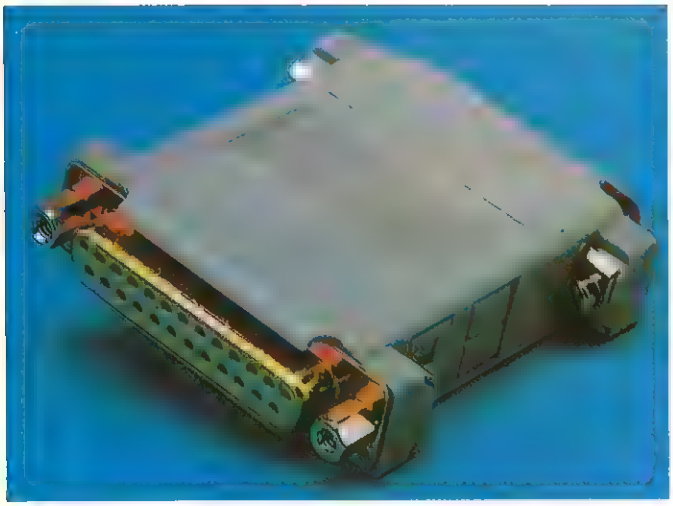
☐ I am interested in exhibiting at Software Tools '91.
☐ I am interested in visiting Software Tools '91.
☐ I would like information on the Conference.

Blenheim Online, Blenheim House, Ash Hill Drive, Pinner, Middlesex HA5 2AE, UK,
Telephone: 081-868 4466, Fax: 081-868 9933, Telex 923498 ONLINE G.

EXE 4/91

BLENHEIM ONLINE
The IT event specialists

CIRCLE NO. 694



How many users of your software paid for it???

SOFTLoK International Limited was established in 1987 with the introduction of our SOFTLoK and SOFTLoK PLUS devices to combat the ever increasing problem of software piracy. Our SOFTLoK range of software protection devices are used by hundreds of software developers from small consultancies to large multinationals. To cope with the ever changing needs of our customers we have developed SOFTLoK II which combines the programmable features of SOFTLoK PLUS with a low unit cost similar to our original SOFTLoK product.

SOFTLoK II™

The Next Generation

SOFTLoK II units are programmable devices containing read/write memory protected by a password. Both the memory and the password can be changed at any time using our routines in your application software. Easy to use menu-driven software is provided to allow small or large batches of SOFTLoK II units to be programmed with their initial data & passwords ready to be sent out with the protected software product.

As SOFTLoK II units plug into the parallel printer port they can be installed or removed in seconds.

Price: 1-19 £16.90, 20-49 £15.40, 50+ £13.70

Evaluation kit £20 (SOFTLoK II, manual & software)
All prices excl. VAT and delivery

- For IBM PC, PS/2 and compatibles
- Uses parallel printer port
- Totally transparent to printer
- Secure data & password can be changed from your application software
- Cascadeable
- 240 bytes of secure read/write memory
- 8 byte (64 bit) password
- No programming adaptors required
- Easy to use SOFTLoK II setup software
- Routines ready to link with various compilers
- Easy to follow manual

I-MEX House, 40 Princess Street,
Manchester, M1 6DE, England,
Tel: 061 228 7379 **Fax:** 061 236 6890

CIRCLE NO. 695

0.0	.5000	
0.5	.6915	
1.0	.8413	
1.5	.9332	
1.96	.9750	(95% around mean)
2.0	.9772	
2.5	.9938	
2.58	.9951	(99% around mean)
3.0	.9987	
3.09	.9990	(99.9% around mean)

Figure 7 - Some entries from standard tables for N (0, 1)

example, `allocated[0]` to `allocated[13]` is 19, `allocated[14]` to `allocated[62]` is 20, and so on. Then we need only generate a random number between 0 and 999, and index `allocated[]` to find a normally distributed random value between 19 and 31.

As a final example, let's consider the algorithm built into `sample()` in the case of a non-normal distribution - say the Poisson Distribution:

((Equation3)) $$\frac{e^{-m}m^{x}}{x!}$$

This curve fits so-called 'rare-events', such as 'phone calls arriving at a switchboard. For instance, to model buses arriving at the bus stop outside my house (certainly a rare event), we first find (by experiment or by consulting the timetable) how many buses we *expect* to arrive in a typical unit time interval - say an hour. Suppose four buses, on the average, arrive every hour. Then we can simulate the situation with the Poisson Distribution P(4). The solution is given in Figure 6. As before, once the groups are allocated, we need only generate a random number to produce a sample that is indistinguishable from a real experiment.

Figure 1, the celebrated Normal Distribution, should be familiar to the most mathophobic reader

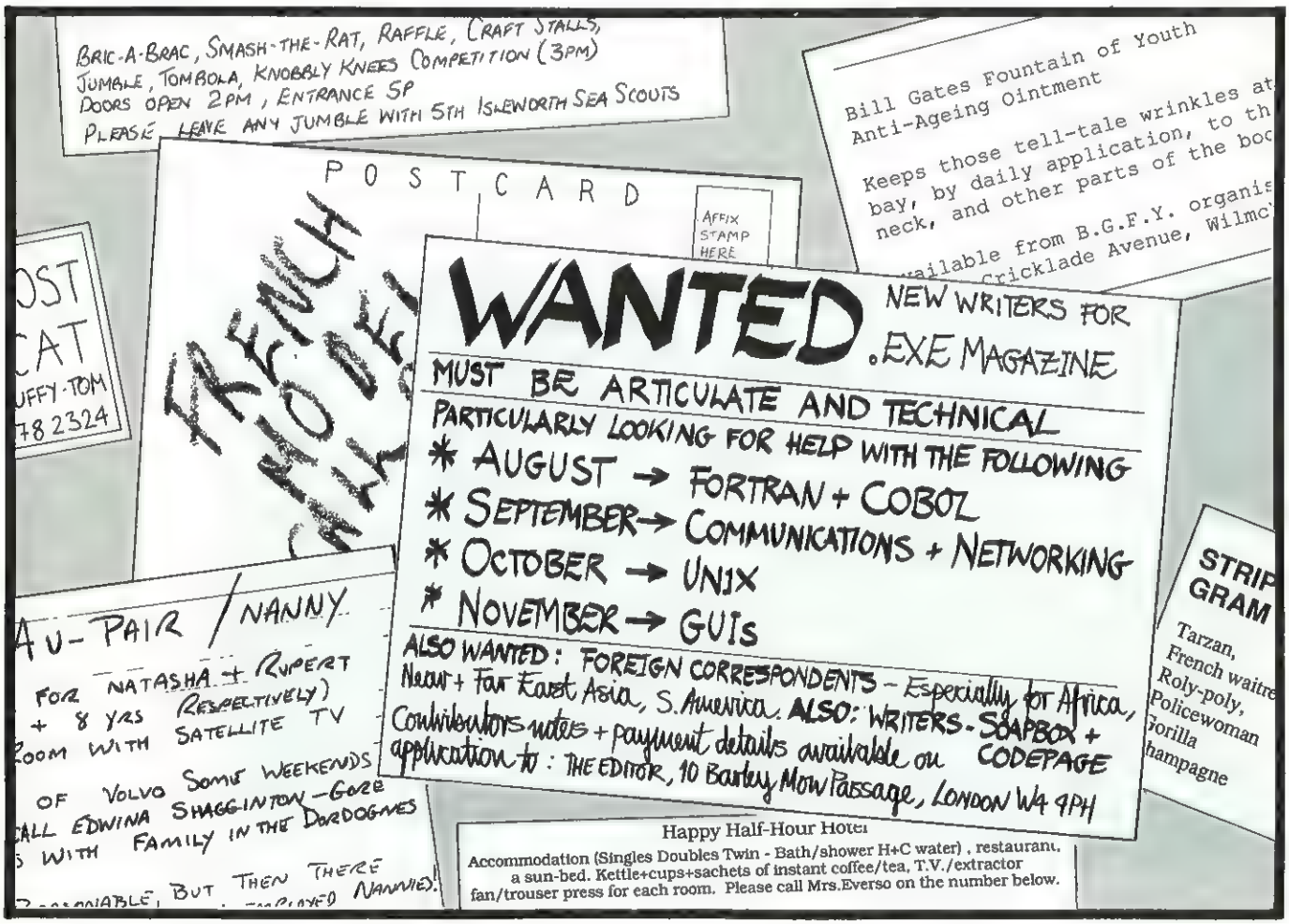
Discussion

According to John Allen Paulos (*Innumeracy*), the science of statistics originated in 18th Century mortuaries, and has never succeeded in throwing off that grim image. In contrast, probability theory was born in the illicit, Bohemian gaming houses of the period, and to this day has a sexy image that has always escaped its less dashing cousin. Certainly, the latter has had a dreadful press; not helped, I believe, by typical uses of the subject in fields such as psychology.

But there is life in the old subject, and used imaginatively it is a vital tool for all scientists and engineers. The fundamental problem with statistics as a science is that it cannot tell you anything you didn't already know; but, like stepping back from a painting, it can reveal truth from what at closer inspection appears mere randomness.

EXE

Robert Loughrey is a writer and engineer living in Sussex. *The author would like to thank John E. for his help in the preparation of this article.*



DOS on UNIX

Our UNIX expert Peter Collinson acknowledges that DOS has its uses, but he doesn't want to clog up his office with a PC, so he has found a software alternative.

Why on earth should you want to run MS-DOS programs on UNIX? Isn't UNIX good enough? Can't you do everything that you would want on UNIX? Everyone seems to pass through a religious period where their system or editor is the only thing worth using in the world. By definition, everything else is complete rubbish. I guess I have got past this with UNIX. I occasionally need to run and develop MS-DOS programs. Also I sometimes want to bridge the gap between the tools that live in the UNIX world and those in the MS-DOS world. These occasions are few, but they happen.

For example, this article is written on a Sun SPARCstation 1+ using a UNIX editor. Our illustrious Editor prefers me to submit things in a form that he can read electronically. So for the final stages of article preparation, I import the text into Microsoft Word running on a program that emulates an MS-DOS machine. I use SoftPC from Insignia Solutions; it runs on a variety of machines and emulates a PC/AT. After some editing, I place the golden words on an MS-DOS format floppy. The floppy is mailed to the .EXE office and injected into the publishing system that is used there (Ventura). SoftPC means that I don't need additional chunks of hardware littering up the place. I can run MS-DOS programs on a UNIX machine.

SoftPC

What exactly does SoftPC do? It's an emulator of an IBM PC/AT containing an Intel 80286 CPU and an 80287 maths coprocessor. It only emulates real mode operation of the CPU - it does not support the 286's protected mode. When the program is started, the emulated machine bootstraps MS-DOS 3.3. It accesses a file in the user's home directory to emulate the battery backed-up CMOS RAM chip installed in the PC/AT. The PC/AT comes with 640 KB of memory and has the ability to add up to 32 MB of expanded memory using LIM Revision 4. This is not part of the address space of the program. It is simulated by creating a file in the same directory that SoftPC is started.

SoftPC runs under the X window system on various platforms. The look of the display on my Sun is shown in Figure 1. I have a monochrome SPARCstation 1+ and run X using OpenWindows, version 2. The example shows the OpenLook window manager, which supplies the border around the display. There are two areas to the window. The large area represents the screen of the PC. The small area to the right contains: the SoftPC logo, some buttons for turning features on and off, and an information area that tells you the keyboard state and what type of graphics emulation you are currently using.

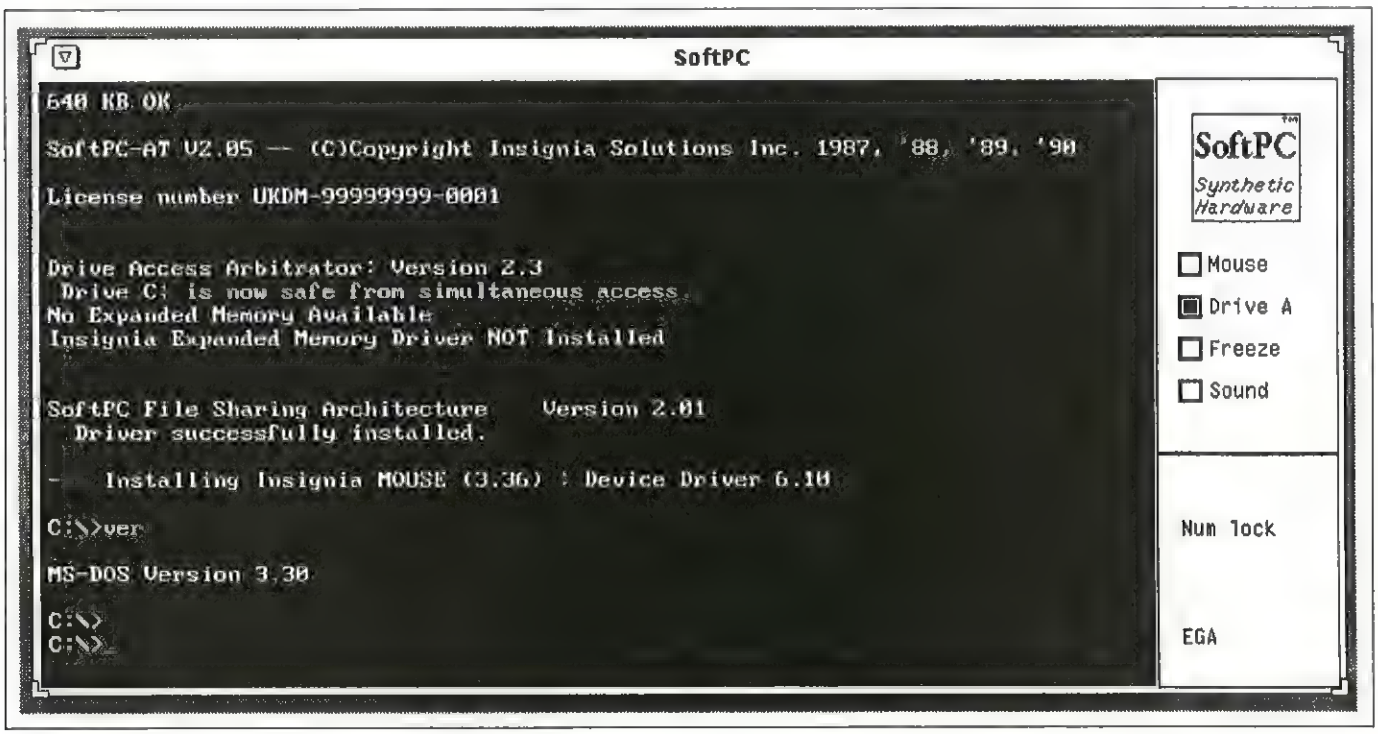


Figure 1 - The SoftPC PC/AT screen just after bootstrap

PROTECT YOUR PROGRAM FROM PREDATORS

- Established for over eight years in software security products
- Internationally established product
- Competitively priced keys (from £18.00)
- Supports IBM PC/XT/AT and PS/2
- Supports Apple Macintosh range
- Advanced manufacturing techniques
- Transparent and user friendly
- No drivers necessary (with exception of programmable key)
- Large range of security keys.

For further information and 60 day evaluation kit, contact:

COMSEC (U.K.) LTD.
Cranfield Airport,
Cranfield,
Bedfordshire,
MK43 0AL
Tel: (0234) 751203
FAX: (0234) 751082

COMSEC
(U.K.) LIMITED
COMPUTER & COMMUNICATIONS SECURITY SYSTEMS

CIRCLE NO. 697

CLIPPER ADD-ON SOFTWARE

QBS Software Limited specialises in providing the best add-on libraries and utilities for **Clipper.** We distribute throughout Europe providing local support centres in France, Germany, Italy and elsewhere through our agents.

FUNCky	General Function Library
Blinker	Dynamic Overlay Linker
Netlib	Networking Library
SilverComm	Communications Library
SilverPaint	Graphics Library
SubNtx0	Filtering Utility
SpellCode	Spell Checker
The Engine	Linkable Spell Checker
Biton	Oracle Library
Scripton	Postscript Library
Overlay 0	Memory Roll Out Utility

For further information, free demo software, prices and how to order please contact:

QBS Software Limited, 10 Barley Mow Passage, London W4 4PH
Tel: 081-994 4842 Fax: 081-994 3441 BBS: 081-747 1979

All trademarks recognised.

CIRCLE NO. 698

System Science

Making Windows

MS Windows SDK
A set of tools and documentation for Windows 3 application development, with a resource compiler, dialog editor, DDE libraries and more.

MS C 6.0
draft ANSI conforming, with editor, code browser, make and link for all development with the SDK and Windows 3.0

save 25% on the new Microsoft C 6.0 and Windows SDK bundle.

WindowsMaker
Windows application design tool and code generator for use with MS C and SDK.

Winpro/3
For prototyping Windows apps. with code generation for resource, header and dialog files.

Case/W
Automate prototyping, design, code generation and apps building for MS SDK and C.

Control Pak/W
Full source code and DLLs for rulers, sliders, toolboxes, picture controls etc, & sample code to illustrate their use.

Growing Windows

Toolbook by Asymetrix
For Windows based hypertext developments with Open Book scripting language and dBASE file access.

Actor 3.0
Object oriented Windows prototyping and development package with debugger and extensibilty with added C routines.

Using Windows

MS Windows 3.0
MS Mouse with Windows
MS Mouse

For Windows

Excel for Windows
Project for Windows
Ventura for Windows
MetaDesign for Windows
Word for Windows
Pagemaker
Corel Draw
Super base 4

3-5 Cynthia Street
London N1-9JF
Fax: 071 -837 6411

VISA
Access

071-833 1022

CIRCLE NO. 699

SoftPC emulates three types of graphic adaptor on the X window screen: EGA, CGA and Hercules. It is easy to switch between the display forms - you make a quick menu selection. This forces the emulated PC/AT machine to reboot, the window resizes and you have a new adaptor in place. I have not managed to use SoftPC running on a colour screen, so I cannot vouch for the colour mappings that are supported. Monochrome screen usage is fine and is actually better that the previous PC emulator that I owned, `vp/ix` running on a Sun386i.

SoftPC can access the floppy in the SPARCstation as PC/AT drive A. Look again at Figure 1. The small square box next to the lettering 'Drive A' is a button that is clicked on by the mouse. The black square means 'on', the drive is enabled and attached to the program. Pressing the button, by pointing the mouse cursor and clicking, will eject the disk and detach the drive from this copy of the emulation. You can attach the drive to another running copy of SoftPC by pointing with the mouse at the relevant button and clicking. If you boot SoftPC with a disk in the floppy drive then it behaves as you might expect: it thinks that the disk is a boot disk and tries to load a system from the floppy.

The 'Freeze' button stops the PC emulation dead in its tracks, and allows you to see what is happening in the machine. The 'Sound' button enables or inhibits the PC's ability to beep at you. The program does not emulate all the sound capabilities of the PC. Finally, the 'Mouse' button affects the mode of the mouse. With the button off, the PC ignores any mouse actions in its display window. The mouse is apparently inoperative if you fire up MicroSoft Word or Windows. Hitting the 'Mouse' button on the SoftPC facia will turn on the PC mouse emulation displaying the appropriate PC cursor shape. Sun machines have a three button mouse, and the left and right buttons map onto the PC's normal two button mouse. The centre button brings up a menu of options allowing you to configure SoftPC and alter its actions - see Figure 2. You can always use the centre button, and you need to. It's the only way of getting the mouse back when SoftPC has grabbed it after you have pressed the Mouse selection area on the facia.

Why on earth should you want to run MS-DOS programs on UNIX?

Disks

SoftPC needs to support a DOS hard disk because many programs expect to find one living at drive C. The hard disk is emulated by creating a MS-DOS file system in a single UNIX file, usually called `hard_disk` on your home directory. The location of this file is configurable, as is its size, which can be up to 32 MB.

You can have two of these MS-DOS file systems in different UNIX files attached as drives C and D should you desire. I haven't bothered. You can also add a RAM disk, which is a big speed win.

In addition, you can make the files in your UNIX file system pop up in the MS-DOS file system by using what Insignia call *File Sharing Architecture (FSA)*. This allows you to associate a directory in the UNIX tree with an MS-DOS drive. The files in the UNIX directory then appear to be placed on the MS-DOS drive when accessed from SoftPC. All normal MS-DOS operations will work on the files, and you can move down the file system using the MS-DOS `cd` command.

UNIX files can be named using any string of characters, but DOS file names are limited to an eight character stem and three letter extension. For things to work nicely, SoftPC needs to convert between these two standards. As it turns out, the conversion is reasonably natural. UNIX files that fit with MS-DOS naming conventions appear with unchanged names. Files that don't are altered so that they are accessible by MS-DOS having a name that resembles the original name. The rules are given in full in the SoftPC installation guide. Files created by MS-DOS are written with a lower-case name on the UNIX directory.

There is one standard FSA connection that is tied to drive E and is configured from the internal setup menu summoned by the centre button. You can add extra FSA drives by editing your `autoexec.bat` to contain lines like:

```
net use h: /home/pc
net use p: /home/pc/pc
```

This attaches drive H to my home directory (`/home/pc`) and drive P to the sub-directory where I store all my PC related programs and data. I tend to keep various compilers and window system on the UNIX disk rather than in a MS-DOS file system, largely because it is easier to delete and compress things.

Moving text files from UNIX into MS-DOS is a bit painful. UNIX ends its lines with a single newline character, while MS-DOS ends its lines with a pair: carriage return and line feed. In addition MS-DOS sometimes likes to place a control-Z character at the end of the file. Insignia supply two programs: `unix2dos` and `dos2unix` that are used to move text files easily between the worlds. You are given both UNIX and MS-DOS version of the programs. It is a little annoying that the UNIX versions cannot be used as filters, taking their input from standard input and writing output to standard output.

Other peripherals

SoftPC emulates two serial devices `COM1/COM2` and two parallel devices `LPT1/LPT2`. These can be connected to files, real devices or pipes. The pipes allow output from SoftPC into programs, so it is easy to make the printer on the emulated PC drive the printer spooler supported on SunOS.

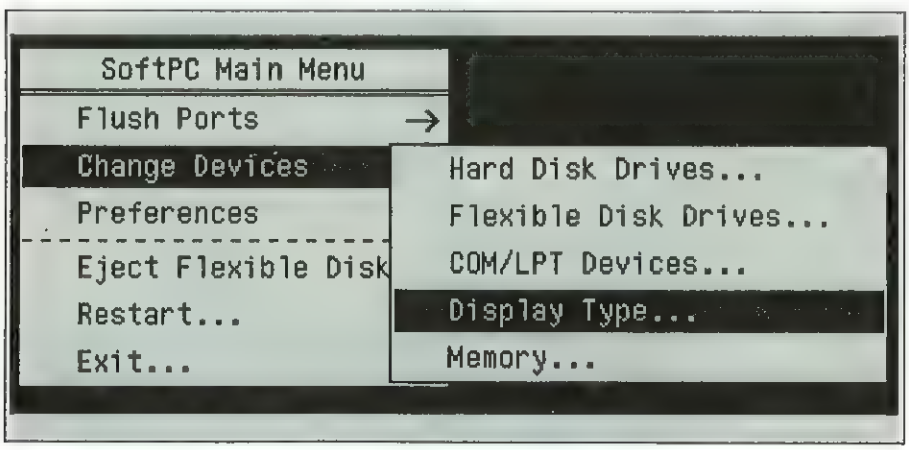
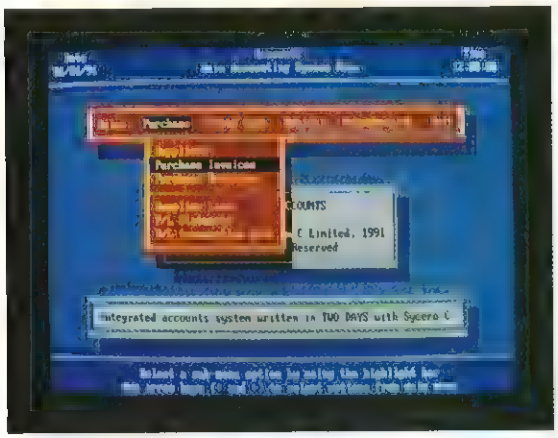
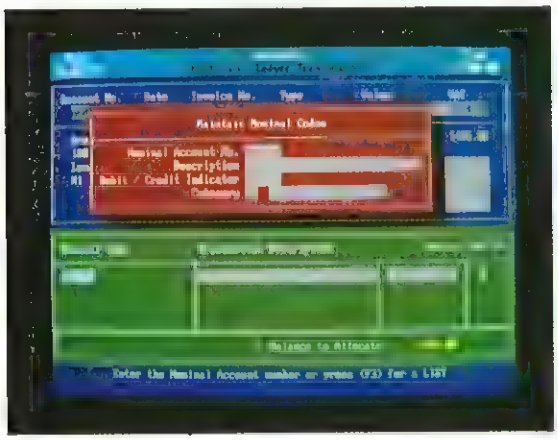


Figure 2 - The SoftPC main menu, summoned by the centre mouse button

SoftwareDevelopers!



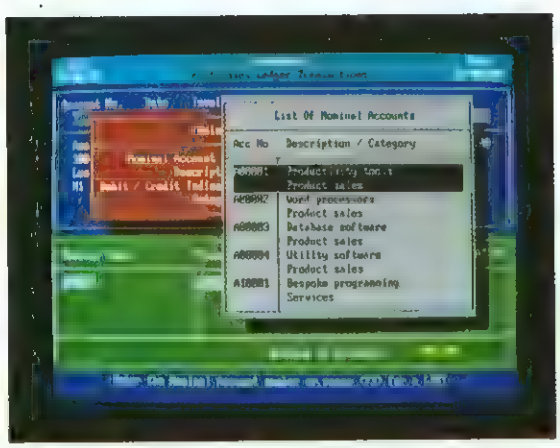
...Here's the same application half an hour later usingTurbo C ++'s advanced VROOM technology...



...and here's the same application an hour later running under SCO UNIX, with no runtime licences required !

Produce a sophisticated application like this in Microsoft C , with either BTRIEVE or C-ISAM, running under DOS in just a couple of days!

...Another twenty minutes and it can access dBASE files...



This entire application was developed in a couple of days using **Sycero C**. Sycero is a powerful program generator that **dramatically** reduces your programming time. Sycero C generates fully structured and documented C source code to be compiled with Microsoft or Turbo C. Sycero C supports the Btrieve/C-ISAM file handling systems, plus it can even generate C programs for use with dBASE or Clipper files and indexes. Sycero C UNIX allows you to develop software to run under SCO UNIX, and you can regenerate existing Sycero C DOS-based applications with absolutely **no changes** required to your definitions.

For more information, either call us on **0622 691616**, or fill out and return the coupon.

CIRCLE NO. 700

dBASE is a trademark of Ashton-Tate. All other trademarks acknowledged

Please send me further details about Sycero C

Name: .....

Company: .....

Address: .....

.....

.....

.....

EXE 3/91

Tel: ..... Fax: .....

I am also interested in your generator for Clipper, Sycero dB ☐

System C Ltd 60-61 High Street Maidstone Kent ME14 1SR

Tel 0622 691616 Fax 0622 691241

It can be a bit fiddly to drive a PostScript printer because most printer spooler software demands a proper PostScript preamble. Also it will not expect to receive the PostScript end of job marker (Control-D). The PC is unaware of these problems and will happily think that it has complete control of the printer. I have found that piping output from SoftPC through the following shell script eliminates most of the problems:

```
#!/bin/sh
(    echo '%!From dos'
   tr -d '\004\032\015'
) | lpr
```

The piece in round brackets prints the string, and then uses `tr` to copy standard input to standard output deleting any Control-D, Control-Z and carriage returns that it finds. All this is fed into `lpr` to send the file to the printer.

Efficient printing is an area of great technical difficulty for the emulator. It is hard for a program to deduce when to close the file and terminate the pipe. The setup menu provides a way of forcing data flush, so the operator can force things to happen.

Installation

SoftPC comes on a single 60 MB cassette. Installation is painless, you just need 6 MB of available disk and a knowledge of how to run `tar`. The installation process comes with a license number that you need to type into a special program.

The 'Freeze' button stops the PC emulation dead in its tracks

You do need to be running OpenWindows 2 supporting an X windows environment on your workstation, but you should be doing this anyway so this is no hardship. The program comes with its own X fonts that you need to inform your X server to use. Finally, there is an X resource control file that is installed in the usual place:

`/usr/lib/X11/app-defaults`.

Each user who will use SoftPC then needs two initialisation files placed in their home directory. The first, called `cmos.ram`, hosts the start-up data for the ROM in the emulated hardware. The second, `.SoftPC` is a file containing textual definitions of the configuration options for SoftPC. This file can either be edited with a normal UNIX editor or SoftPC will change values from menu selections. Once the standard files are in place, then you can go. Type

```
SoftPC
```

and you should see something very much like Figure 1.

Problems

I haven't encountered too many problems. But then perhaps I haven't stressed SoftPC too greatly. It has run all the programs that I have tried to use with no difficulty, and at an acceptable speed. The Norton Utilities' SI program ran correctly, giving SoftPC a score of 8.0 for its Computing Index and 10.7 for hard disk speed - this with no other significant UNIX tasks running on my Sun SPARCstation 1t. However, mouse handling is a little sluggish.

Upgrade Your IBM XT Clone To An AT 286!

Make your machine run up to 11 times faster!

By allowing us to replace your old eight slot XT clone motherboard with a new AT 286 compatible board running at 11 times the speed of an average XT.
That's almost twice the speed of many AT's!

We will fit a special 286 board with eight expansion slots to allow the use of your old eight bit XT cards. (This includes hard disk controller cards which can cause problems with normal 286 boards)

NOTE: This is not to be confused with accelerator cards that give only a fraction of the performance and on average cost more than double the price of our upgrade.

This upgrade is fitted with 512K RAM as standard, 640K or 1Mb RAM as an optional extra.

Also possible are upgrades to 286 with 16 bit expansion slots, 386 SX, 32 bit 386 and 486 processor boards.

If you have an IBM XT or compatible that is in need of repair, why spend money on trying to revive it when we can not only get your machine working, but fit it with a 286 motherboard!

We will upgrade an 8 slot XT compatible, to a 286 fitted with 512K RAM for just:

£200 + VAT & Carriage
Please telephone for details:
(0424) 721761

KJD Software Systems

Miracle Menu Ver 2.00
* Requires no overhead memory.
* Runs programs and DOS commands.
* 15 Menu pages with a further 15 options on each.
* Automatically returns to the relevant menu after executing a program.
* Passwords can restrict access to menus, options, setup programs & DOS.
* Assignable "Buzz words" ie call your favourite program "Fred" & from then on type "Fred" anywhere in the menu structure to execute it.
* Options can be selected using all popular ways, including highlight bars, direct access etc.
* Supports all directory & drive changes.
* Full manual & Help page included.

Miracle Labels Ver 1.50
* Stores names, addresses, postcodes, phone, fax & telex numbers, vendor rating plus additional comments.
* Can cope with almost any size label and has multi-web facility.
* Print multiple copies of labels.
* Hard disk install & installation to "Miracle Menu" programs included.
* Efficient retrieval by vendor rating, postcode or name.

Miracle Invoices
The complete invoice management system.
* Discount facility on each item.
* Keeps database of customer details for fast creation of invoices & can share "Miracle Labels" database.
* User definable printer functions.
* Invoices can be marked as paid.
* Summary & statement facilities.
* Up to 99 lines of description on each item.
* Up to 99 items per invoice.
* Allows customisation of page layout for use with plain paper & pre-printed stationery.
* Hard disk install & installation to "Miracle Menu" programs included.

All packages for just £39.95 each
(VAT + P&P Inc.)
All Software available on 3.5" & 5.25"
Please state which.
SPECIAL OFFER
SAVE £30!
When you buy any 2 packages together for just £49.95

Credit Card Orders
Telephone HASTINGS
(0424) 721761 (24 Hrs)

Unit 10 Mercatoria Business Centre
100-102 Norman Road
St Leonards-On-Sea

CIRCLE NO. 701

Snow Reports

Calling all systems houses...

Your search for a fast, flexible and sophisticated report generator, query facility and applications interface has just ended. You have discovered SNOW.

Development companies which are seeking to enhance their software products should take a close look at SNOW. They will soon realise that the sheer power of SNOW REPORT WRITER can add a great deal of value to their system.

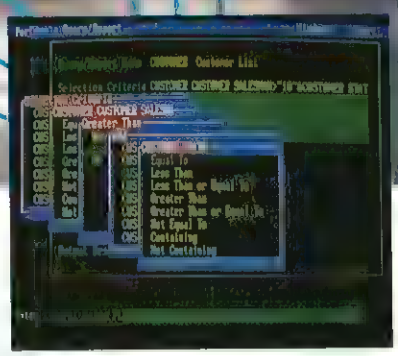
And talking of great deals we also have a superb package deal in store for SNOW resellers. It could mean some very attractive margins and greater sales for you. Fully supported training and distribution advice are just part of the marketing back-up you'll get as a SNOW reseller.

In report generation, SNOW is the shape of things to come so why not capitalise now on the chance to boost both your software products and your sales. Your first step is to fill in the coupon or call Melaney-Ann Ward on 0800-833374 and we will send you your free information pack.

In a high pressure Environment

Snow

EUROSNOW UK



Please send me details on SNOW and how we can join your re-seller network **SA30**

Name: _____

Position: _____

Company: _____

Address: _____

_____

_____

Telephone: _____ UNXS _____

Send to: EuroSnow UK, Thomas Bennett House, Salford University Business Park Salford M6 6AJ

CIRCLE NO. 702

CopyControl

THE NEW GENERATION OF COPY PROTECTION

NO	User hassle
NO	Back-up problems
NO	Hardware add-ons or special disks needed
NO	Changes to source code required
YES	CopyControl beats ALL bit-copier Programs
YES	Floppy disks, hard disks and networks supported
YES	CopyControl is totally transparent to the user
YES	You can limit program use by no. of copies, no. of executions or date
YES	CopyControl works on all IBM compatibles
YES	Free demo disk available

For Further Information Phone or Write to

microcosm

Microcosm Limited,
17 Cranbrook Road, Bristol BS6 7BL
Telephone: 0272-441230

CIRCLE NO. 703

F77L-EM/32

Port 4GB mainframe applications to 80386s with this 32-bit DOS-Extender compiler. The winner of PC Magazines's 1988 Technical Excellence Award just got better. New Version 3.0 and OS include: Editor, Make Utility, Virtual Memory Support, DESQview Support, New Documentation and Free Unlimited Runtime Licenses. F77L-EM/32 and OS/386. POA.

F77L

The compiler of choice among reviewers and professionals. Includes a Debugger, Editor, Profiler, Linker, Make Utility, Weitek and 386 Real-Mode Support, Graphics. POA.

Lahey Personal Fortran 77

New Version 3.0: Full ANSI 77, Debugger, Editor, Linker, Library Manager, Microsoft and Borland C interfaces, 400 page Manual. POA.

System Science

3-5 Cynthia Street,
London N1 9JF
Tel: (071) 833 1022
Fax: 081 837 6411

CIRCLE NO. 704

With Highland Add The Power Of Graphics!

DYNAMITE DEMOS with GREAT GRAPHICS

Prototype & Demo System

Now you can demo your applications with Highland's PictureBox in almost any graphics mode including SuperVGA 800x600x256 or any of the standard VGA modes.

Use the integrated development system and the powerful Zsoft PC PaintBrush IV Plus included to create, organize, edit, and debug your demo.

Use special effects like Wipes, Explodes, Fades, Splits, Diagonals, etc. on the PCX bitmaps or on the text screens.

All commands can be controlled through mouse driven 3D button menus for intuitive and intelligent program design.

No Royalties +More

Supports 30+ graphic modes plus text, with special effects; cut, move, size & scale images; text on graphics; simple to use - no programming; 250+ page manual with tutorial; build fully interactive demos that respond to keystrokes, data entry, timeouts and mouse events, and much more.

Provided is a free runtime version you can distribute with your professional demo.

PictureBox Only!

£149.95

Order Today!

PCX *Effects & Text* IMAGE Solutions!

VBase Colour Database £249.95

Turn your dBase, Clipper, C, Basic, Pascal, application into a powerful 256 - COLOUR DATABASE with little programming skills. Includes full printer support ! Most any application can be converted in just a few hours.

PCX Programmers Toolkit £165.00

Includes over 75 routines to display, save, scale, and print PCX bitmapped graphics from almost any programme. Full use of virtual buffers and EMS 4.0.

PCX Effects £85.00

Add spectacular special effects (F/X) to your program. Wipe, split, crush, slide, sand, drip, diagonal, spiral, random, or explode your graphics. Requires PCX Toolkit.

PCX Text £149.00

Display blazing bitmapped text in any graphics mode as simple as text mode. You can display your text information on top of graphics.

GX Graphics £179.95

A complete graphics library supporting all graphics primitives. Use the GX Graphics Toolkit instead of the BGI or MS libraries and make your program faster, smaller and more portable across compilers - while supporting more video modes.

Pinna Electronics Ltd., APL Centre, Stevenston, Ayrshire KA20 3LR

Tel: (0294) 605296 or Fax: (0294) 68286

CIRCLE NO. 705

You do need to recompile the SunOS kernel to include the System V IPC facilities, adding the lines

```
        # SV IPC message
options    IPCMESSAGE
        # SV IPC semaphore
options    IPCSEMAPHORE
        # SV IPC shared memory
options    IPCSHMEM
```

into the configuration file for your machine. Insignia's documentation doesn't tell you this.

The main problem is that SoftPC doesn't seem to tailor into the UNIX X environment very nicely. It works fine, but some things are a little painful.

For example, it tends to grab the mouse and not let go until you use the main selection menu to separate SoftPC mouse function and normal X operations. I feel that when you move outside the screen display area then the mouse should revert to its action in the window it is currently pointing to.

Also, the program also turns off keyboard auto-repeat. This is intrinsically no problem. However, the keyboard is a global resource. I do feel that a program turning keyboard repeat off should at least turn it back on when the program exits, which SoftPC doesn't. My current SoftPC command resets auto-repeat on exit from SoftPC by using the script:

```
#!/bin/sh
/h/SoftPC/SoftPC
xset r
```

You can also add a RAM disk, which is a big speed win

SoftPC tends to drop files in places that are not controllable by the user. For instance, the `cmos.ram` file must be on the user's home directory. Also, the file that shadows LIM memory is simply placed in the directory where SoftPC was started. This can be really inconvenient - I want to be able to say where it should be put.

SoftPC is big, around a 6 MB process image. My machine has a lot of memory and swaps from a local fast disk. The program starts reasonably speedily, I can imagine that it will be a mite slower if you are loading over NFS.

Overall assessment

All the problems above are small ones. I think that the product is reasonable. No-one is suggesting that you replace your PC hardware by SoftPC, if you have a lot of PC work, then simply buy a real machine. But for occasional use it works well and is usable. I think that running Aldus Pagemaker inside MicroSoft Windows 2 is a fair test of the emulator's capability. This runs just fine. All in all - a nice product.

EXE

Peter Collinson is a freelance consultant specialising in UNIX. He can be reached as `pc@hillside.co.uk` *electronically (although your mailer might be happier to put the address the other way round) or by phone on 0227 761824.*

SoftPC for SPARC is distributed by Technology PLC (0925 830404) and costs £595. You can get information describing the other versions (including Apple Macintosh and NeXT) direct from the manufacturer Insignia Solutions (0494 459426).

X11.4 for UNIX System V 386 can now support

32,768 colours !

Upgrade to 16 Million Colours soon!

Also Video Capture & Display using X Windows

Graphics Cards supported:

EGA, VGA, SVGA, 8514, Targa 16, Targa+, Hercules Graphics Station Card.

Runs on the following UNIX Operating Systems:

InterActive 386/ix, SCO UNIX, ESIX.

All Trademarks acknowledged

Powerful UNIX Systems for the serious developer

£ 3,990

386-33 with cache
8 MB memory
120 MB, 18ms Hard Disk
1.2 MB Floppy Drive
2 serial, 1 parallel port
14 inch **VGA Monitor**
256 colours Graphics Card
Tower Case with 275 W power supply
UNIX System V
X Windows version 4

Call us now for your custom configuration

UniVision (UK) Ltd

PO Box 1098, Swindon, SN5 6LP

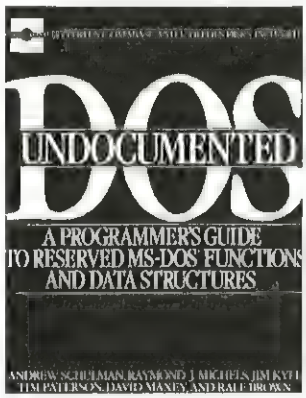
tel (0628) 822281 fax (0628) 822580

CIRCLE NO. 706

Books

Geoff Chappell takes a close look at an important DOS book, and David Gristwood buries his head in UNIX manuals.

Spreading Secrets



Many DOS programmers will have found that some tasks simply cannot be programmed using only the information contained in technical references. DOS has a secret side and programmers who know it can perform seemingly impossible tricks. It is intrinsic to the nature of secrets, however, that access to them is not easy, so to quite a few programmers, even serious DOS experts, a 700-page book titled *Undocumented DOS* will seem like manna from heaven.

Programmers who need to know about unlisted functions, mysterious structures and flags or just more detail on how DOS accomplishes its normal work have had to rely on the scattered, vague and often contradictory material published thus far - an unhappy situation indeed, which this book sets out to remedy.

For the most part, *Undocumented DOS* confines its scope to the central components of DOS. To DOS programmers, DOS is the Int 21h interface when seen from inside their programs and the command processor outside. System resources are examined from DOS's perspective to see how the information gained may be put to good use in applications.

However, the book is a lot more than a list of undocumented DOS functions and examples of their use. The reader (assumed to be fairly knowledgeable and treated as such throughout the book) is guided through programming techniques not often touched upon. Assembly language is used sparingly, the opportunity being taken to show how almost every topic (even TSR programming) can be implemented in a high-level language.

Still, whatever its objectives and style, a book on this subject, claiming to give detailed descriptions of DOS's undocumented functions and structures, is open to assessment on its comprehensiveness and the reliability of the information it contains.

The book's annotated list of undocumented DOS functions is an extract from the very much larger 'Interrupt List', well-known to the on-line community and described in the book as 'the one truly definitive, absolutely reliable list of DOS calls'. The full list is supplied in a hyper-text version on one of the 1.2 MB disks which accompany the book.

An objective assessment of the list's completeness is a little tricky, since the subject is to some extent open-ended. In those cases in which DOS calls other programs, it may simply be impossible to deduce everything about such calls from the code in DOS alone. Quite reasonably, the greatest effort has been concentrated on DOS's core and readers must expect to see the word 'unknown' when the book touches on matters outside its primary scope.

Against this, it must be remarked that there are several complete omissions even from the Int 21h interface and that the easy majority of unknowns can be resolved by inspection of only the DOS kernel and programs supplied with it as standard (particularly SHARE and IFSFUNC). Although only some contributing authors raise the matter of disassembly, it has certainly taken place. How else can it be known for instance that the unknown argument to Int 2Fh function 122Fh is required in register DX? Why else are those fields in the Swappable Data Area which are incremented or decremented in the DOS code described as 'unknown flag or counter' rather than just 'unknown'?

That the book should mention the Swappable Data Area at all, much less attempt to detail it and illustrate its effective use in TSRs, is very welcome and demonstrates that *Undocumented DOS* is more comprehensive than anything which has gone before. Also welcome is a discussion of the SEGDEBUG interface and documentation on network redirectors, a topic said to be mysterious even within Microsoft. Even so, the information given on redirectors is best viewed as exploratory, for it is too vague to be much help in writing a serious installable file system - it misses the attribute word at offset 0Ch in the IFS driver header, for instance.

Of course, omissions and incompleteness are not likely to cause harm. The authors are commendably frank about the limits to their knowledge and imbue a proper sense of caution. DOS features which are largely unknown are not going to be incorporated into tomorrow's software without further investigation by developers and the book does a good job of pointing the way, even providing a script-driven utility for spying on interrupt communications.

Much more concern must be given to the reliability of some descriptions of DOS's inner workings. Descriptions of memory allocation strategy and of actions taken to open a file contain errors of fact (although they may be popular misconceptions). According to the book, when DOS opens a file, it searches for a free entry in the handle table belonging to the current process and then for a free System File Table.In fact, it looks through the SFTs first - a difference which is not trivial, being the scurce of a DOS bug not mentioned in the book. Similarly, the distinction between a device driver and file need not be delayed as late as the book asserts; DOS's pathname parsing routine will search the device driver chain in some circumstances (the source of a bug which is mentioned in the book) and this is even shown in another chapter by the conversion of A:CON to A:/CON.

Perhaps it is unfair to labour such points, given a generally fine book, but precisely because *Undocumented DOS* will be regarded as a standard reference for its subject matter, it is important to note that it has lapses in reliability. This is nowhere more apparent than the several bugs which afflict the DEVLOD program for loading device drivers from the command line. In short, because its author does not seem to understand the reason behind one of the fields in DOS's List Of Lists, using DEVLOD to load ANSI.SYS from the command line has the possibly undesirable result that Ctrl-C and Ctrl-Break will not be effective when programs have their standard input redirected from a file. For much the same reason, combined with a misuse of undocumented function 32h, the ability of DEVLOD to load block device drivers in the presence of SUBSTed drives is suspect, despite claims. Thus, on a system with hard disk partitions C: and D: and a drive E substituted for a directory on drive C, using DEVLOD to load a block device driver (as drive F) is very unpleasant for drive D and is not a success for F: either.

More subtle is the construction of double-checks based on observations that are not supported by careful analysis. The clearest example is the run-time determination of System File Table size to allow verification against the size known from the DOS version number. The book assumes that the SFTs will be ordered AUX then CON, but this need not be true, for the arrangement can be manipulated by software using only documented DOS functions (though a good reason may be hard to produce).

Such problems aside, however, *Undocumented DOS* is a serious, responsible study which succeeds on the whole in demonstrating the safe use of DOS's secret side to expand programming horizons.

Title: Undocumented DOS *Author: Andrew Schulman, et al*
Publisher: Addison Wesley *Price: £39.95*
ISBN: 0-201-57064-5 *Pages: 700*

ADDISON-WESLEY PUBLISHERS

THE LATEST BOOKS ON DOS

UNDOCUMENTED DOS
A Programmer's Guide to Reserved MS-DOS Functions and Data Structures
A Schulman et al

At last, the information that IBM programmers have been waiting for since MS-DOS first appeared - a complete guide to the MS-DOS functions that Microsoft has left undocumented, reserving them for its own use in software development. This example-rich reference offers comprehensive coverage of all the reserved MS-DOS functions. The accompanying disk contains an online database of all DOS functions and a powerful monitoring utility that allows programmers to view the actual code as they compile their programs. Topics covered include TSRs, DOS multitasking, networks, shells, and debugging.
Book/Disk Package/600pp/0 201 57064 5/paper £35.95

EXTENDING DOS
R Duncan et al

This book will help bring all the power of a new generation of PC hardware to your DOS applications. It is a complete and authoritative guide to all aspects of extending and enhancing DOS. It will show you how to address memory above 640M with EMS and XMS standards, how to take advantage of the speed and increased addressable memory of 286 and 386 architecture, understand industry standards such as VCPI and DPMI, and implement operating environments such as Windows and DESQview. With each chapter written by an acknowledged expert on that topic, it provides the most comprehensive information available on extending DOS.
1990/448pp/0 201 55053 9/paper £20.95

UNIX

UNIX SURVIVAL GUIDE
Tim Parker

This is the *'everything you ever wanted to know about using UNIX'* book. This detailed, step-by-step guide leads you through every aspect of working with UNIX, from turning the system 'on' to becoming adept at all of its essential features. The book offers clear explanations on such essential topics as logging on and off, background processing, the mail program, word processing with the ed and vi editors, and system security. Completely up-to-date, this book covers the most recent release of UNIX, System V, Release 4, and the major graphical user interfaces: OPEN LOOK, OPEN DESKTOP, and X. It is an easily accessible introduction for the first time user and a great source for users switching to UNIX from other operating systems.
Spring/304pp/0 201 57078 5/paper £20.65

THE ULTIMATE GUIDE TO EMACS
UNIX Text Editing
W Arnold et al

This is an excellent step-by-step introduction to Emacs, UNIX text editor. The book explains all the parts of the EMACS screen, working area, status line, command and message line, and all key commands and special keys are explained and defined. Three separate chapters are devoted to using EMACS to write C, FORTRAN and Lisp programs. The book includes a simple explanation of a text editing session so that the reader can understand how to start EMACS, create a buffer, type and save text, edit and print text. It also contains an overview of EMACS programming concepts that includes language models, syntax-directed editing, indenting, commenting, compiling, debugging and documenting variables.
Spring/250pp/0 8053 0621 8/paper TBA

OSF/MOTIF

OSF/MOTIF™
Concepts and Programming
Thomas Berglage

Covering versions 1.0 and 1.1 of OSF/Motif, this book has been written by an experienced applications developer to provide a thorough understanding of this popular toolkit. Special features of the book include:

- **a clear explanation of the concepts and underlying technology**
- **detailed coverage of the features of OSF/Motif not readily available in reference manuals**
- **numerous examples demonstrating how to extend Motif functionality and interface Motif code with an application**
- **practical advice and helpful hints for avoiding the pitfalls commonly encountered by Motif programmers**

Professional programmers, application designers and consultants will find this book an invaluable reference on OSF/Motif.
February/501pp/0 201 55792 4/paper £24.95

X WINDOWS

X WINDOWS PROGRAM DESIGN AND DEVELOPMENT
Steven Mikes

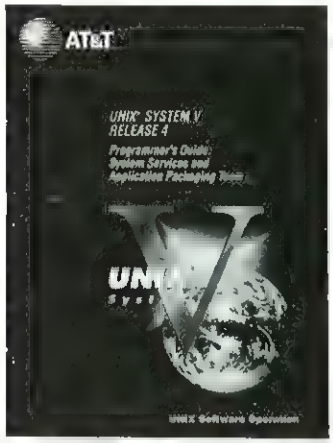
The X Window System is the de facto standard development environment for creating UNIX applications. This book offers programmers advice from an established UNIX authority on designing efficient and elegant programs with the most recent release X11.4. The author reveals tips and techniques from the real-world experiences of many professional X Windows programmers, and demonstrates how to accomplish important programming tasks with actual code examples. The book addresses the practical issues faced by every X Windows programmer in designing and developing working applications, enabling the reader to acquire the skills necessary to become an expert programmer.
1990/496pp/0 201 55077 6/paper £24.25

For further information, please request a copy of our 1991 Complete Computing catalogue from:

ADDISON-WESLEY PUBISHERS
Finchampstead Road, Wokingham, Berkshire RG11 2NZ
Tel: 0734 794000 Fax: 0734 794035

CIRCLE NO. 707

Manual Entry



The first implementations of AT&T's UNIX System V Release 4 - the latest and the most powerful version of the UNIX operating system - are just starting to appear. Combining elements from a number of UNIX variants, such as Microsoft's XENIX, Sun Microsystem's Sun OS, and the BSD system, SVR4 has looks to be industry standard UNIX platform for the 1990s.

Prentice-Hall, in conjunction with AT&T, has produced a series of books to accompany the release. The full series comprises a staggering fifty titles, and covers most aspects of UNIX, including system administration, programming, and networking. As well as providing a separate series of titles for the Intel 80386 specific versions of the operating system, there is a graphics series covering both X Windows and Open Look, and a binary interface and compatibility specification set for a range of processors, including the Intel 80386, Motorola 68000 and Sun SPARC.

The books in the series are designed to compliment the documentation supplied by AT&T, and so all the information in the Prentice-Hall books can actually be found in the official AT&T documentation. These books will, however, appeal to large corporate users and training companies, as well as technical departments, who usually need multiple copies of technical material such as this. The books are by no means inexpensive - the four in this review alone cost over £100, but, for your money, you do get a weighty set of documentation, and bulk purchases of over 30 copies are eligible for discount.

The four books reviewed here are part of the general programming set, and can be viewed as a foundation series for programming under UNIX. They are not, though, the only technical books in the series, and programmers may find themselves having to refer to additional books, depending on the nature of the work. Although these books are aimed at generic UNIX, the AT&T 3B2 and 6386 computer systems are used as examples throughout the text, and, therefore, developers working with the Intel 386 family would be best advised to go for the 386 specific titles where possible.

The *Programmer's Guide: ANSI C and Programming Support Tools* is an introduction to the UNIX C compiler and the supporting programming tools. It is not an introduction to the C language, for it assumes some degree of proficiency in C programming on the part of the reader, as well as a basic understanding of the UNIX environment. The book concentrates on the compiler and linker, and using a tutorial style approach, the reader is taken through the basic set of operations in a 'try it and see' format. The latter part of the book is devoted to the utilities that accompany the development environment, including the symbolic debugger, the profiler, and source code control system, as well many of the well known UNIX utilities, such as `yacc` and `lex`.

The tutorial style used throughout the book works well, and the writing style remains relaxed and informal. If anything, the text sometimes verges on the verbose, which unfortunately tends to obscure, rather than clarify, particular points. Despite this, it contains just about all the information needed to get to grips with the UNIX C compiler.

The *Programmer's Guide: System Services and Application Packaging Tools* builds very much on the previous title in its coverage of programming under UNIX, and accordingly assumes a fair degree of fluency in the C language - and, again, some basic knowledge of UNIX principles. The emphasis of this book is on actually writing programs to operate under UNIX SVR4, and most of the book is devoted to an in depth examination of the system services and facilities available. The topics covered include file handling, memory management, process control, and inter- process communication.

The material here is more technically oriented, and this is reflected in the heavier reliance on source code extracts. There is also, unfortunately, a degree of terseness in the text, which makes reading heavy going at times. One other criticism that can be levelled at this book is the lack of full programming examples, a somewhat surprising state of affairs for a book so clearly aimed at programmers.

The essential volume for UNIX programmers is the *Programmer's Reference Manual*, which contains most of the information programmers will probably ever need to know about UNIX commands, C library routines and systems calls, not to mention file formats, and other miscellaneous details. Old time UNIX hackers will quickly recognise the MAN format used - the information is divided into short individual entries, one for each command, system call, structure and subroutine, and these are listed alphabetically, grouped into individual sections. Each of the entries is typically one or two pages long. Each one contains a brief synopsis, a fuller textual description, along with examples, notes, references or diagnostic information that may be relevant.

Once familiar with this style of presentation, it is surprising how well it works. The brevity and rigidity of the form restricts longer explanations - particularly in the coverage of the C library, where code examples are rare, and at most three or four lines. The only other problem is trying to hunt down information about a particular topic or subject matter, rather than a specific command or library routine. Fortunately there is a permuted index, in which all the significant terms are listed with a line long context, and page reference. This allows fast access to material, particularly when searching through multiple index entries.

Finally, the *POSIX Conformance* volume is a much slimmer affair of 50 pages, aimed at C programmers who need to write application programs that conform to the IEEE Portable Operating System Interface (POSIX) standards. In line with the requirements of the POSIX standard, this document outlines the implementation-defined features, as used by UNIX SVR4. It comprises two main sections. The first simply lists the values, limits and function prototypes for the header files `unistd.h` and `limits.h`, whilst the second section covers the implementation dependent features of the POSIX standard under UNIX SVR4. The resulting text consists of simple annotations to the P10003.1 standard, arranged in order of their appearance in that work. You'll probably need the standard by your side, since what is left consists almost entirely of tables and implementation statements. It's a specialised book, for programmers who need to work with the POSIX framework.

Title: Programmer's Guide: ANSI C and Programming Support Tools — *Author: AT&T Labs*
ISBN: 0-13-933706-7 — *Publisher: Prentice-Hall* — *Price: £32.20*

Title: Programmer's Guide: System Services and Application Packaging Tools — *Publisher: Prentice-Hall*
ISBN: 0-13-947060-8 — *Price: £27.85*

Title: Programmer's Reference Manual — *Publisher: Prentice-Hall*
ISBN 0-13-947029-8 — *Price: £33.95*

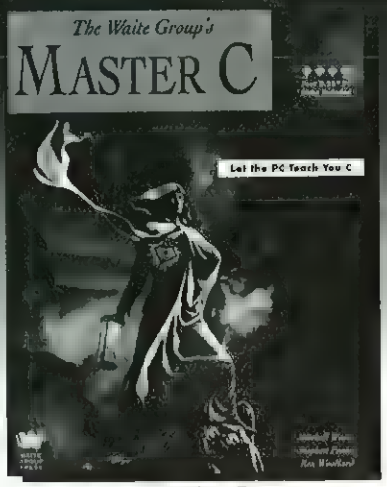
Title: POSIX Conformance — *Publisher: Prentice-Hall*
ISBN 0-13-933672 — *Price: £11.30*

Books Received This Month

Turbo C++, A Self-Teaching Guide by Bryan Flamig	John Wiley	£19.95	ISBN 0-471-52903-6	pp375
The Oberon System by Martin Reiser	Addison Wesley	£24.95	ISBN 0-201-544229	pp350
Intelligent Embedded Systems by Louis L. Odette	Addison Wesley	£36.85	ISBN 0-201-517531	pp270
Novell Netware System Overview by Novell	Addison Wesley	£28.75	ISBN 0-201-57027-0	pp346

POWER PROGRAMMING BOOKS BY MAIL ORDER

	C	£
4836	C Programming Language 2nd Ed	26.95
4482	Learn C Now BkDk	* 35.94
5752	Learning to Program in C	4.95
4728	C A Dabhand Guide	14.95
6612	Teach Yourself C	19.95
7455	Waite Group C Bible 2nd Ed	27.50
5640	Teach Yourself C	21.50
4928	C Quick Reference	7.95
7730	Using C	27.45
6194	C Complete Reference 2nd Ed	26.95
1137	Big Red Book of C	8.95
6534	C Step by Step	27.50
5275	Graphic Programming in C	24.95
7529	Network Programming in C BkDk	* 46.64
6137	C Programmers Toolkit BkDk	37.51
7137	Waite Group C Program 2nd Ed	25.95
7174	Waite Group New Primer Plus	24.95
7906	Small C Compiler BkDk 2nd Ed	* 23.30
3428	C Prog Guide to Serial Comms	27.95
5133	C Programmers Guide to NetBIOS	27.50
5267	C Traps & Pitfalls	14.40
8618	Encyclopedia C	35.95
8497	Waite Group Master C BkDk	39.94
4047	C Power Users Guide	19.95
4272	Programming in C Revised Ed	27.00
8168	C Puzzle Book 2nd Edition	32.10
7105	C Programming Complete Guide C	19.95
7712	Introducing C	12.95
5047	C Pocket Reference	5.95
	C - QUICK C	
6855	Quick C Program Guide	22.95
7840	Quick C Complete Reference	22.95
5532	Waite Group Quick C Bible	27.50
	C - TURBO C	
5416	Waite Group Turbo C Program PC	20.95
4354	Turbo C Complete Reference	24.95
7999	Using Turbo C 2nd Ed	21.95
5722	Mastering Turbo C 2nd Ed	20.95
	C++	
6234	C++ Primer	22.95
2887	C++ Programming Language	21.95
7716	Teach Yourself C++ BkDk	32.14
7550	Programming Zortech C++	16.95
6118	Programming in C++	19.95
6189	Using C++	22.95
6439	Object Oriented Programming Des C++	20.95
8749	C++ Toolkit	19.95
6801	Object Oriented Environment C++	26.95
	C++ TURBO C++	
6956	Waite Group Turbo C++ Bible	27.95
6613	Turbo C++ Complete Reference	22.95
6687	Using Turbo C++	22.95
6955	Waite Group C Prog Using C++	27.50



Master C

With a superb interactive Training Disk, this package is the best tutorial on C available to date.
Waite Press
Ref:8497 *Price: £39.94*

	PASCAL	£
3739	Illustrating Pascal	11.95
8096	Pascal by Example An Intro	12.95
7713	Teach Yourself Pascal BkDk	* 31.62
7806	Mastering Pascal Programming	5.99
	PASCAL - TURBO	
5482	Turbo Pascal Quick Reference	7.95
5464	Using Turbo Pascal	20.95
8838	Turbo Pascal Complete Ref 2nd Ed	22.95
6997	Master Turbo Pascal 5.5 3rd Ed	27.50
6847	Turbo Pascal Internals BkDk	* 47.15
8838	Turbo Pascal 5.0 Complete Ref	22.95
5634	Turbo Pascal Advanced Tech	20.95
6396	Master Turbo Pascal 5.0 2nd Ed	22.95
7691	Turbo Pascal 5.5 Programming	19.95
6197	Turbo Pascal Disk Tutor BkDk	* 38.06
8013	Up & Running Turbo Pascal 5.5	8.95
7956	Graphics Prog Turbo Pasca. 5.5	22.95

	BASIC	£
8000	BASIC Prog Inside & Out BkDk	* 33.44
8041	BASIC 2nd Ed	18.95
	BASIC - GW BASIC	
5891	GW-BASIC Prog for Beginners	14.95
6249	MS GW-BASIC Users Guide & Ref.	12.95
6850	Using BASIC GW-BASIC & BASIC A	18.45
6919	First Book GW-Basic	15.50
6547	Program in GW-BASIC	14.95
8129	Teach Yourself GW-BASIC	14.95
5751	Program in GW BASIC	9.95
5660	GW BASIC Made Easy	19.95
3842	GW BASIC Quick Prog Ref Guide	8.95
7542	ABCs of GW-BASIC	18.95
7839	GW-BASIC Reference	22.95
	BASIC - QUICK	
4910	Using QuickBASIC 4	20.95
7196	Waite Group QuickBASIC Bible	24.95
5633	QuickBASIC Advanced Techniques	20.95
5904	MS QuickBASIC Quick Prog Ref	5.95
6446	MS QuickBASIC Version 4.5 3rd Ed	18.95
7195	Waite Group QuickBASIC Primer	22.95
6513	Learn BASIC Now BkDk	35.95
	BASIC - TURBO	
3761	Using Turbo BASIC (McGraw-Hill)	19.95
4664	Using Turbo BASIC (Sigma)	11.95
6441	Turbo BASIC Prog Tech	18.45

The latest computer books by mail order

021 706 6000

CM
COMPUTER MANUALS

Computer Manuals Ltd. Dept 282
30 Lincoln Road Olton Birmingham B27 6PA
Tel: 021 706 6000 Fax: 021 706 3301

Call and order now with credit card or send cheque with order. Full catalogue available on request, over 3000 titles in stock

UK P&P £3.25, free when you order 4 or more books. Prices marked * include VAT. Prices are correct at time of going to press but are subject to change without prior notice

CIRCLE NO. 708

G.W. Computers Inc
ADC1, 4 Eagle Square,
E.Boston. MA02128
U.S.A.
Tel: 617-569-5990
Fax: 617-567-2981

Xvale Ltd., Store 'D'
55, Bedford Court Mansions,
Bedford Avenue, London.
WC1B 3AD U.K.
Tel: 071-636-8210
Fax: 071-255-1038

These 'bundled' systems are probably the best deals available. If you only want hardware then just call us for a quote!!

***** LINX 8088 -10 *****

PC computer with 640k ram, 20mb disk, Mono monitor, 101 keyboard, 2S/1P/1G, 1 * 5-1/4" and 1 * 3-1/2" drive
+++ £695 .00 includes £270 FREE software +++
Menuix & Sysix Database software £99.00
Sysed programmers editor £49.95,
G.U.R.U. Reporting utility £49.95
Gawk & Grep prototypers £35.00
IXpro screen designer £35.00

***** LINX 80286-16 *****

PC computer with 2mb ram, 40mb disk, Mono monitor, 101 keyboard, 2S/1P/1G, 1 * 5-1/4" and 1 * 3-1/2" drive
+++ £1020.00 includes £370 FREE software +++
Menuix & Sysix Database software £99.00
Sysed programmers editor £49.95,
G.U.R.U. Reporting utility £49.95
Gawk & Grep p £35.00 , IXpro screen designer£35.00,
Cshell Unix environement and tools £99.00

***** LINX 80486-25 *****

PC computer with 8mb ram,200mb disk,SVGA 1024 * 768 Colour scm, 101 keyboard,
2S/1P/1G, 1 * 5-1/4" and 1 * 3-1/2"drive, 4 Port Unix serial , Lnx Ethernet , Mouse
£3695.00 includes £1730 FREE software
Unix SysV ,rel 3.2 ,V2.02 Workstation developer £1295.00
Comprising: R-time,Vpix, Software developement, NFS, TCP-IP, XWindows RT & Developer Ten-plus u/i,
(VPIX is DOS-Merge under UNIX) Unix Text-Processing system £195.00
Menuix & Sysix Database plus IXpro designer £134.00 Sysed programmers editor £49.95 G.U.R.U. Reporting utility 49.95

***** LINX 80386sx-20 *****

PC computer with 4mb ram, 80mb disk,
VGA 1024 * 768 Colour monitor,
101 keyboard, 2S/1P/1G,
1 * 5-1/4" and 1 * 3-1/2"drive, mouse
+++ £1760.00 includes £800 FREE software +++
Unix SysV ,rel 3.2 ,V2.02 Rtime & Vpix £576.00
Menuix & Sysix Database software £99.00
Sysed programmers editor £49.95
G.U.R.U. Reporting utility £49.95
IXpro screen designer £35.00

***** LINX 80386-33 *****

PC computer with 4mb ram, 120mb disk,
SVGA 1024 * 768 Colour scm, 101 keyboard,
2S/1P/1G, 1 * 5-1/4" and 1 * 3-1/2"drive
+++ £2535.00 includes £1300 FREE software +++
Unix SysV ,rel 3.2 ,V2.02 Rtime & Vpix £576.00
Unix software developement system £495.00
Menuix & Sysix Database software £99.00
Sysed programmers editor £49.95
G.U.R.U. Reporting utility £49.95
IXpro screen designer £35.00

CIRCLE NO. 709

'C' PROGRAMMERS

Advance your career into DIP – The fastest growing sector of the worldwide computer market.

to £20,000 + Bonus + Benefits Yorkshire

As the Document Image Processing market advances, Image Systems Europe are widely acknowledged to be the No. 1 UK company, and are well positioned for further expansion in meeting the technical and commercial challenges of the future.

We now need additional programmers who have gained at least 12 months experience in 'C' and MS Windows within a PC environment. Familiarity with ORACLE/INGRES or other SQL based rdbms' is advantageous but not essential as comprehensive technical and product training is provided.

You'll be working in a small, friendly development team who recognise and employ new technologies to develop DIP systems across a diverse range of applications.

Excellent working conditions and starting salaries of up to £20,000 are on offer. Company benefits are also exceptional including: an INCENTIVE BONUS • FREE BUPA • NON CONTRIBUTORY PENSION and RELOCATION PACKAGE to a particularly attractive area of low cost quality housing.

'Enhance your image' by forwarding your CV (Ref. No. C804) to **Andre Evan Smith** at: **Aardvark Swift Associates Ltd, Westcourt House, 75-77 Station Street, SWINTON, South Yorkshire, S64 8PZ**

Alternatively, please contact **Debbie Leighton** on **(0742) 420419** for further information.

IMAGE SYSTEMS EUROPE LTD

Exciting opportunities in

REAL-TIME SOFTWARE

*** TRANSPUTERS** — **Essex to £25K**

If *you* are a Software Professional with experience in signal processing or parallel processing then this (UNPARALLELED!) OPPORTUNITY could be for you.

*** ADA** — **Berks & Middx to £25K**

We are currently recruiting high-calibre engineers experienced in ADA/VAX/VMS Structured Methods for a number of prestigious projects. Excellent opportunities, big company benefits.

*** C++** — **South to £22K**

Our client, an international company providing ATE solutions is offering excellent prospects to Software Engineers with experience of UNIX/Windows/C

For further information about these and our many other opportunities telephone:

0727 41101

or 0727 30602 (out of hours)

Alternatively fax your C.V. on 0727 838272

95 Victoria Street, St. Albans, Herts AL1 3TJ

JOHN PROCTER RECRUITMENT

PRE SALES SUPPORT
LONDON & BIRMINGHAM
£27K PLUS CAR

Mini/UNIX, Office Automation, Networking. Corporate Clients. No finer time to join.

SOFTWARE ENGINEERS
WORCS, COVENTRY & NORTH BIRMINGHAM
£13K TO £18K

Real-Time Industrial/Automation. C/UNIX, Graphics, Windows.

ORACLE/ACCELL/UNIFY
INGRES/INFORMIX/VAX/VNS?
WEST MIDS, WARWKS, WORCS, STAFFS.
£12K TO £27K PLUS CAR.

Vacancies from Programmer through to Project Manager. Please Post/Fax your C.V.

COBOL OR C WITH UNIX
CHESHIRE
£13K TO £25K PLUS CAR

Positions available at all levels in Development, Support and Q.A. X/Open, Windowing, Commercial Systems.

SOFTWARE SYSTEMS ENGINEERS
HEMEL HEMPSTEAD
TO £22K

P.C. UNIX/C. Device Drivers. Netware. Structured career path.

IBM/AIX EXPERIENCE?

If you are seeking a position in this area or whatever level in the West Midlands contact me now. My internationally based client needs you.

SOFTWARE/FIRMWARE ENGINEERS
BRACKNELL
TO £24K

P.C. UNIX/C Real-Time, Automation, LANS Graphics, Networks,Device Drivers. Commitment to quality is this company's hallmark.

SOFTWARE ENGINEERS & SNPS
BRISTOL
ALL LEVELS TO £18K

Real-Time Multitasking C on PCs for technical processes.

ABACUS CAREERS LTD

3rd Floor, Smithfield House, Digbeth, Birmingham B5 6BS. Tel: 021-666 7484. Fax: 021-666 7430

Looking For A Software Challenge...??

Look at GIS

Geographical Information Systems (GIS) are one of the fastest growing application areas of the 90's and the UK is at the forefront of core software developments. GIS exploits all the emerging technologies including UNIX, networking, WIMP, graphics and innovative database and software engineering approaches, including object-orientation.

Below are just FOUR of the numerous opportunities on offer.

Software Engineer, £16000, N. Home Counties.

Suit a recent Computer Science graduate with a 2:1 or better. Research into innovative graphics and data management! Train on spatially-referenced systems.

FORTRAN Programmer, £20000, S.W. London.
Suit a PC enthusiast with simulation/modelling applications experience. Train on windows, 'C' programming and 4GLs.

ORACLE Programmer/Analyst, £22000, Thames Valley
Suit a bright problem solver looking for consultancy. VAX/VMS experience a help. Train on GIS applications.

UNIX/'C' Software Engineer, £24000, Between M4/M40
Environmental Executive Information Systems development! Learn OOP, X-Windows and MOTIF.

If you want to know more about software opportunities in GIS call ALAN CARNELL, at Concurrent Appointments, the GIS recruitment specialists.

Concurrent Appointments
Software Recruitment

27 FIELD CLOSE
HARPENDEN
HERTS
Tel: 0582 712976
Fax: 0582 764858

CAD ENGINEER (2)
BERKS & SURREY £20K+
Both require good CAD skills; one invergraph and one traffic applications. Good companies and usual benefits.

HARDWARE ENGINEER (DEVELOPMENT)
KENT & SURREY £ Good
ASICS Low level voltage etc - good microprocessor experience - hands on essential: interesting applications - world leader.

SOFTWARE ENGINEER
BERKS £18-20K
WINDOWS - C - COBOL - good development projects: GUI & OOPS interest essential.

COMMS PROGRAMMER
W. LONDON/SURREY £18K
COMMS - DOS - LSICOMMS - PROTOCOLS - X25 - C - & lots & lots of Connectivity.

GOOD C AND WINDOWS
SILICON VALLEY £LOTS
A proven ability, good track record, blue chip development: a rare opportunity.....!

REGENCY
COMPUTER SERVICES

EUSTON HOUSE, 81-103 EUSTON STREET
LONDON NW1 2ET FAX: 071-439 4972

071 388 2312 (24 HRS)

COBOL/CICS Analyst Programmer to £20k Package
A major institution based in the city is seeking an Analyst/Programmer with a minimum of 2 years COBOL programming in a CICS environment under VM-DOS/VSE. The ideal applicant will have good knowledge of JCL and Easitrieve. This is a new position in a new department and will require considerable application in the initial stages. Long term prospects are available to the right applicant.

UNIX/C Programmer to £20k
A software house in Central London developing products for the financial sector is seeking an additional 2 programmers with solid experience of 'C' under UNIX. A knowledge of development under X-WINDOWS and using LISP expert tools would be a distinct advantage. The positions would be mainly development but some degree of program maintenance and customer support would be required. Good prospects are available to the right applicants who have that extra zest of professionalism.

Technical 'C' Programmer to £18k
A UNIX software house based in Hertfordshire who specialise in solutions based around accounting and donation software running on IBM RS/6000 or NCR Tower are seeking an addition to the programming team. The ideal candidate will have a minimum of 2 years 'C' programming technical, as opposed to application solutions under UNIX & AIX. A good knowledge of hardware related programming, communications and specialised interfaces is expected. This is a dynamic company with a good future.

'C' Programmer with Tax experience £Neg
A specialised organisation in Sussex is seeking a 'C' Programmer, under DOS & NOVELL, with a solid knowledge of Tax and accounting procedures to assist in the development of a bespoke package. The appointment will initially be on a 6 month contract basis but could be renewable for a further term. The organisation have invested in some of the latest program tools which will be available for use by the successful applicant.

Programmer/Analyst to £21k + BB
A major clearing bank is seeking a programmer/analyst with a minimum of 1 years ORACLE experience preferably in a VAX/VMS environment. Knowledge of Pro-Cobol and DBA would be a distinct advantage. The ideal applicant will have good communication skills, be self-confident and be able to use their own initiative. The position would suit a technical person who wishes to develop their analysis skills and supplement their IT skills as their career progresses with this prestigious organisation.

For details of these and other positions contact Nick Hodson on the number below or out of office hours on 081-390 9518

RONICOM
RECRUITMENT

46-47 Pall Mall, London SW1Y 5JG
Telephone: 071-321 0245 Fax: 071-839 7629

If your skills are in the frame we can help you find a position.

Opportunities North of Watford
contact
Actis Recruitment
Tel: 0204 20200
17 Chorley New Road
BOLTON
BL1 4QR.

*Unix is a trademark of AT&T

ACTIS
RECRUITMENT

WEST YORKSHIRE

Programmers (particularly COBOL and 4GLs), Analyst Programmers, Software Engineers, Project Leaders, Support (pre and post sales), Trainers (MS-DOS, UNIX and Applications). Also Hardware/Software Design Engineers.

FOR YOUR NEXT CAREER MOVE AROUND WEST YORKSHIRE

Telephone Vincent Atherton on Leeds (0532) 504560 or write to:

AIREDALE RECRUITMENT

Realtex House, Micklefield Lane, Rawdon, Leeds LS19 6AX

AIREDALE RECRUITMENT

Information Systems Group Plc.

ISG plc

Special System Group

The Special Systems Group is a team of programmers developing PC/Unix based communications software for use within the corporate market. Its aim is to produce programs that are simple to use, yet highly sophisticated in functionality.

We now have opportunities for Programmers with a minimum of two years' "C" experience under MS-DOS. Familiarity with 80x86 assembler, Windows 3, and Unix would also be of great advantage. We particularly seek a high level technical ability and creativity, joining a team facing some of the most professionally challenging development projects.

Salary £16,000 - £20,000 + bonus

To be considered for the above position, applicants should send a C.V. to Marcus Poulton at the following address.

Information System Group
The Old Forge, Chipstead,
Sevenoaks, Kent TN13 2RY
Telephone: (0732) 457001

CAD CAM CONCEPTS 3

Despite the recession, well-qualified technical and commercial professionals are still in demand, especially with experience in one or more of the following areas:

SOFTWARE APPLICATIONS DEVELOPMENT
GIS *or* finite element analysis *or* nc

SYSTEMS SPECIALISTS
Unix support or X-Windows/GUI tools development - or object oriented techniques

COMMERCIAL SUPPORT ENGINEERS
product marketing *or* nc post sales support/customisation.

If you would like to know more about these or other vacancies, please send your cv to **Sheila Fenlon or Michael von Rimscha, CAD CAM CONCEPTS (Recruiting), Campbell House, Campbell St, Cambridge CB1 3NE or telephone us on 0223 412965**

40,000 Software Professionals want to know about your company!

OK I can't guarantee they all do at the same time - but there's one way to find out:

Advertise in the next 2 issues of .EXE

MAY **Includes Software Development News** - the preview magazine for Software Tools '91 - Europes largest Software Event.

JUNE The official magazine of **Software Tools '91** - 5,000 extra copies being distributed at the show.

The count-down for Software Tools '91 is under way

To take advantage of the increased circulation of .EXE magazine call Sandra Inniss-Palmer or Ed Butcher on

081 994-6477

and hear about the best advertising opportunity you will receive in 1991

.EXE Magazine

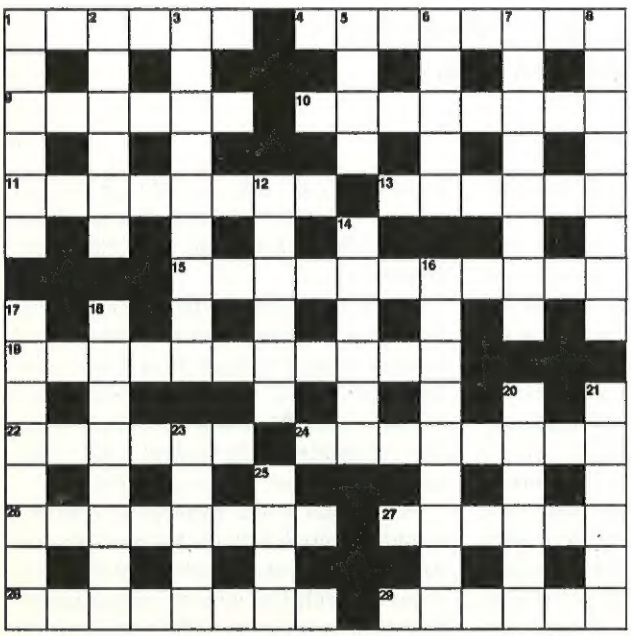
Probably your only choice.

Editorial Index

If you are interested in the News articles on pages 4, 7 & 8, and would like to receive more information about the products mentioned, please circle the corresponding circle number on the Reader Service Card at the back of this magazine.

COMPANY	PRODUCT	CIRCLE NO
Bits Per Second	dGE 4.1	715
Blackwell Scientific Publictns	WinBasic	716
Borland	Borland C++	717
BSO Tasking	Real Time Executive/386	718
Computer Solutions	Catalogue	719
Defiant	Virus Hunter	720
Digital Research	Multiuser DOS	721
DRA	WordStar	722
Glockenspiel	CommonView	723
Hitex	C51 User Group	724
HP/Sun	OMF	725
Interactive	Unix V.4	726
Microsoft	Pen Windows/PDS	727
Mirabilis	LISTPACK+	728
MultiScope	Windows Debugger	729
OMG	Object Management Guide	730
Peripheral Land	SuperFloppy 1.4	731
QBS	Scripton	732
Solution Systems	Charge	733
System C	Sycero C for Unix	734
SystemStar	dbVista V3.2	735
Watcom	Watcom C/p16	736
WDA	Windows Development	737
WordPerfect	DOS Developer's Toolkit	738
Ziff Communications	Computer Select	739

.EXEWORD APRIL

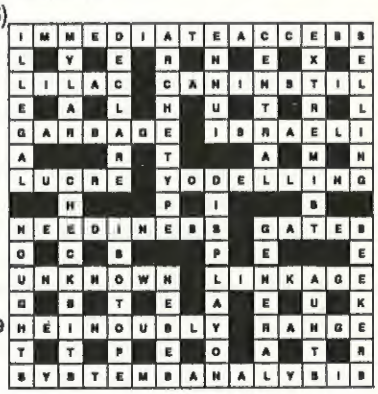


ACROSS

1 Keys to the simplest conditional (2,4)
4 Get together the source statements? (8)
9 Obtain second opinion on input (6)
10 Check out terminal in the bottom of the castle (8)
11 Repeats something about 21 (8)
13 Like Lady Diana on Saturday somewhere (6)
15 Practical appliance of knowledge, re software to us (11)
19 Remarks on printout (11)
22 Story of New Testament shows real skill (6)
24 Declare, and find yeti somehow (8)
26 Record presence in very immediate cell (8)
27 Symbol for aluminium in the message (6)
28 Lard fuel? No way - that's awful (8)
29 Bits of bits in a parallel process (6)

DOWN

1 Get NOTed! (6)
2 FOR x = 1 to 3 should do it (6)
3 Such a program does things right (9)
5 Network at the top of the bill...(4)
6 ...outside as well (5)
7 Comms board with a shot inside (8)
8 Twilight working time - not for odd people (8)
12 Crews from software houses? (6)
14 Save? (6)
16 Totally necessary as petrol in France may be (9)
17 Got data by force? (8)
18 Good model of wavy 27 (8)
20 Link between IT and molecular engineering? (6)
21 Units of machine action on the road (6)
23 Sensed information as child rose over the man (5)
25 Mouth-watering choice? (4)



.EXEWORD MARCH

ADVERTISERS INDEX

ADVERTISER	PRODUCT/SERVICE	CIRCLE	PAGE
Addison Wesley	Book Publishers	707	89
Bits Per Second	Graphics Server Software	674	50
Bits Per Second	Graphics for dBASE	693	76
BL Security	Software Protection	685	61
Blenheim Online	Software Tools Exhibition	694	77
Blinker	Clipper Fast Linker	656	21
Borland	DOS and Windows Programming	663	33
Borland Competition	Competition	648	6
Brent Communications	MAX copy PROtection	677	51
Camel	Plotter Utilities	676	51
CEBRA Communications	Multi VGA Adapters	679	52
Computer Bookshops	Computer Books Distribution	708	91
Comsec	Software Copy Protection	697	81
CTL	Copy Protection Hardware	684	61
DES	Software Protection	681	54
EIUG	Informix User Group	696	64
Eurosnow	UNIX Report Generator	702	85
Expert Systems	Prolog for Windows	652	17
Glockenspiel	C++ Windows 3 Library	655	19
Grey Matter	Programming Tools	646	3
Highland Graphix	Graphics Menu	660	29
Intasoft	Software Management System	653	17
IXI	UNIX Software and Training	712	64
Jensen & Partners	C++ Compiler	670	41
KJD	IBM PC Upgrade	701	84
Lahey	FORTRAN Compilers	704	86
Linx I	All UNIX Systems	691	70
Linx II	All UNIX Systems	709	91
LPA	AI/KBS/OOPS Software	666	39
Magnifeye	Software Protection Device	672	47
Microcosm	Copy Protection Software	703	86
Microft Technology	Network Security	680	53
Microsoft I	C Compiler	661	27/29
Microsoft II	Windows 3.0	683	56/57
New Dimension Technologies	Multi-Platform xBase	664	34
Nu-Mega	Debugging Tools	651	14
Pecan	The Pecan Power System	658	25
Pinna Electronics I	Tools & Graphics	705	86
Programmer's Odyssey	UNIX Software	711	40
Prospero	DOS Extender Kit	670	44
QA I	Network Training	645	IFC
QAII	4GL Development Tools	687	63
QBS	Clipper Add-ons	698	81
Rainbow Technologies	Software Data Security	647	5
Real Time Software	CASE for Windows	654	17
Recital	RDBMS/4GL for VAX & UNIX	692	75
Roundhill	Development Tools	669	42
Salford	FORTRAN for DOS and UNIX	668	39
SCL	Communications Boards	688	64
Select Software	CASE Tools	667	39
Sequiter	C Library	682	55
Softlok International	Piracy Protection	695	78
S/ware Construction Company	Development Tools	671	45
S/ware Generation	Code Generator	657	22
S/ware Paradise	Business Software Sales	690	68/69
Solution Systems	Programming Editor	665	37
System C	Program Generators	700	83
System Science I	Development Tools	686	61
System Science II	Windows Tools	699	81
System Science III	Version Control System	675	50
System Star	dbms Four C	649	9
Univision	UNIX Development Systems	706	87
USA Software	Programming Tools	662	30
User Friendly	Software Copy Protection	678	52
Vleermuis	C++ Class Tree OS/2	650	11
Watcom	C Compiler	710	OBC
Zortech I	Multi Plat C++ Comps	673	49
Zortech II	C++ Video Tutorial	689	67

STOB - Upgrade

The latest upgrade from Will 'n' Dave's Software Ltd, WADSbase V3.2, has arrived at Stob's offices.

I'm sitting with the girls in the 'canteen' (sink, kettle, one spoon, plus an old, broken toaster that the MD brought in because he thought one of us might like to 'try and get it going'), mulling over who is the most fanciable TV cop. Trace has already poured scorn on Julie's suggestion of Morse ('but he's so *old*, and he's a slob'), and Julie's reciprocated by crushing Trace's nominee of Nighthawk ('a slime-bag, and he's never home nights'). Both now warming up to have a go at *my* sweetheart, Spender (old, slob, slime-bag, thinks he's a rock star), when Mike comes in with a parcel for me.

Adjourn debate, and return to my office clutching prize. Package proves to contain the new wonder-release of WADSbase, with added compiler RAM-packing CUA SQL OOP active liposome extensions that will revolutionise your code and help prevent the signs of ageing. There are three ring-bound manuals, an envelope containing six x 1.2 MB flops, a set of dividers and a good handful of polystyrene petals just ripe for stuffing down Julie's neck. Rip open envelope (thus implicitly agreeing, in the event of unauthorised duplication, to consign my unborn children to slavery) and take permitted single backup of disks like you are supposed to, for oh! what a good girl am I. Assemble manual components. This is the bit the magazine reviewers always whinge about, rendered complacent by much too much free software. Ah, the excitement of slitting open the Cellophane, the smell of the unthumbed pages, the wonderful sentences that come from automatic spell-checking of technical English ('...do not attempt to decompose your interpretives...'), the discovery of three Section 2s...

Place extra Section 2s in death-to-pirates envelope, disk 1 in cake-hole B:, run INSTALL.EXE. Much drive churning, followed by multicoloured screen containing more questions than a poll-tax evasion form. Do I want to overwrite previous version of WADSbase? Yeah, what the hell. This completed, hit ENTER, feed in disks as requested, fine. Edit CONFIG.SYS and remove the 'FILES=20' line that has been inserted beneath the old line 'FILES=20'. Ready to go.

Switch to the Johnson directory, compile the report program as shown in Section 2 number 1, run it, lock up. Boot, back to the directory, edit the source. Complete source file now reads ◆£a*beep*!#ë, which is not how I remember it. Excitement at new compiler rapidly turns to dust, then fury.

Think: what would Spender do? Spender would call up WADS on his portable and abuse them in witty Geordie. WADS having come up with the wheeze of putting its Technical Support on 0898 (peaktime 48 pee per grunt) number, and such numbers having been rendered mechanically undialable from office phones by our toaster-manager, this option not open.

Julie puts her head round the door to say What about Bergerac's bottom; 'cos Trace says he's put on weight recently. Test polystyrene petals. Petals work fine.

EXE

Opportunities for Software Professionals

SYSTEMS PROGRAMMER

BERKSHIRE **To £20k**

1-2 years COBOL experience within software tool development or applications programming. Knowledge of PC DOS & OS/2 with presentation manager. To provide product development & support skills for important COBOL tools. *Ref: 03/91/AJW*

SYSTEMS PROGRAMMER

BERKSHIRE **£13-15k**

To provide programming & maintenance support for end-user products in the IBM PC market. Experience of 8086/ 80286/ 80386, Assembler, DOS or OS/2 for creation of COBOL code generator

Ref: 03/91/BJW

SENIOR SOFTWARE ENGINEER

HAMPSHIRE **c£18k**

Experience of real-time programming in C & Assembler with knowledge of MS-DOS & UNIX. Conversant with use of logic analysis & in-circuit emulators. To join a team engaged in design & development of software modules for use in fixed control applications

Ref: 03/91/CJW

SENIOR ANALYST PROGRAMMERS/ ANALYST PROGRAMMERS

BERKSHIRE **£14-20K**

Part of project teams covering system development (Software, hardware & telecommunications & training, communications, procedures, documentation). IBM 3090/300J mainframe environment running CICS under MVS/ESA operating system with DB2 database via X25 network. CSP used for development work, FOCUS for report writing. Minimum 2 months experience as analyst/programmer on IBM platform. Degree qualified.

Ref: 03/91/GJW & HJW

DOCUMENTATION TECHNOLOGY PROGRAMMER

BERKSHIRE **£Neg**

Responsible for designing & programming components for workbench product to enable users to create documentation & tutorials from applications. Products include graphical user interfaces & object orientation. Would suit COBOL or experienced programmer with interest in text processing/DTP. *Ref: 03/91/DJW*

SOFTWARE DESIGNERS TELECOMMS

SURREY **£15 - 25k**

Design of software for real-time embedded applications. Knowledge of structured techniques and CASE tools eg MENTOR, together with programming experience in 'C' would be an advantage.

Ref 03/91/NJW

SOFTWARE PROG/ANALYSTS AS400/IBM SYS 36/38

MOST AREAS **TO £25K**

We have many clients seeking experienced software engineers who have worked in an AS400/IBM mainframe environment. Positions range from programmers through to senior analysts and support personnel.

Ref 03/91/OJW

PRODUCT LEADER

BERKSHIRE **£21-28k + Car**

To be responsible for all aspects of system development of the job management system, including project planning & control. Liaison with variety of users & external suppliers. Present development areas include MIS & interfaces between JMS & Ether corporate systems. Sound project management skills required. Familiarity with mainframe systems essential.

Ref: 03/91/IJW

SENIOR SYSTEMS ANALYST

BERKSHIRE **£22k - 24k**

A degree plus a minimum of 8 years experience desirable in civil engineering systems. CSP/FOCUS experience, Artemis/AS project management system. IBM mainframe environment, PC's & or prime systems. IBM 3090 environment with MVA/ESA operating systems. Also use of CICS, TSC, & COBOL2 & GPG.

Ref: 03/91/JJW

SUPPORT CONSULTANT

BERKSHIRE **£Neg**

Responsible for providing a range of support services to users, resellers & internal sales. Expected to be a source of expert knowledge on the use & application of time specific product range. Experience of post-sales support, minimum 4 years PC environment plus MS-DOS, Paradox, Lotus 123 (or Quattro) PC networks, especially Novell & 3COM, UNIX or VAX systems.

Ref: 03/91/KJW

SYSTEMS PROGRAMMER

BERKSHIRE **TO £17K**

Create new Run Time Systems using 8086/80286/ 80386 assembler and provide maintenance support to all OEM customers. Additional experience of three of the following is essential:- DOS, OS/2, COBOL, C, FLEX, UNIX or VM.

Ref: 03/91/MJW

KNOWLEDGE ENGINEERS

HOME COUNTIES

Rapidly expanding Expert Systems Software house requires young developers to design and implement projects. 1 year's experience minimum of any AI language or C, and UNIX etc.

Ref: 03/91/LJW

For further information on these or related positions call
JEREMY WILLAN or SARAH HOLTHAM on
(0734) 774234 or (0604) 33195 after 7p.m. FAX: (0734) 772773
Or write in confidence to CPS at:-

63 Peach Street
Wokingham
Berkshire RG11 1XP

CPS
COMPUTEC PERSONNEL SERVICES

WATCOM C8.0/386
Optimizing C Compiler and Tools
for 386 Extended DOS

WATCOM C/386
for Windows

Unleash 386 Power on Your Microsoft C Code.

- Interactive source-level debugger
- Generates high-performance code for 32-bit protected mode
- Microsoft source and library compatible
- Fast, tight code
- Profiler
- Protected-mode version of compiler
- Graphics library
- 100% ANSI C and SAA compatible
- Run-time compatible with WATCOM FORTRAN 77/386

Experts Agree on WATCOM C:

"When Novell went looking for a 32-bit compiler for use with the NetWare 386 developer's kit, the company selected WATCOM's...It's clear that Novell chose wisely; this product is a winner."
Fred Hommel, BYTE, December 1989

"WATCOM C/386 is a fantastic new ANSI C compatible compiler for 386-based PC's...If you have written your application in Microsoft C, you will love this compiler."
J. Richard Hines, Electronic Test, December 1989

"Microsoft library- and source-compatibility makes WATCOM C7.0/386 ideal for porting DOS applications to 32-bit native mode. This compiler enables full 386 performance without 640K limitations."
Richard M. Smith, President, Phar Lap Software, Inc.

"WATCOM is definitely the leader in object-level optimizations...For flat-out executable speed... WATCOM C showed shining performance."
Computer Language, February 1989

WATCOM C8.0/386 Professional

• 100% ANSI C optimizing compiler • Protected-mode version of compiler • 386 run-time library object code • Windowed source level debugger • Profiler • Editor • 386 graphics library • MAKE • Linker • Object-code librarian • Object-code disassembler • Supports Phar Lap and ERGO DOS extenders

WATCOM C/386 for Windows

- Enables 32-bit Windows 3.0 GUI applications
- Interactive debugger for 32-bit Windows GUI applications
- Ideal for porting 32-bit Unix applications to Windows
- 32-bit flat model simplifies Windows memory management
- Royalty-free run-time license

0 SDK,
extende

1-800-265-4555

.EXE
Please mark 'Later' if appropriate
Notify 050 SAH for address change

		Date Seen
~~EC DISPLAY 2 WKS~~	~~LIBRARY, EC~~	16.4.91
~~PJ WINFIELD~~	~~INF/TECH, EC~~	
~~DG MARR~~	~~INF/TECH, EC~~	
EC LIBRARY	EC	

COMPUTER LANGUAGE PRODUCTIVITY AWARD 1990

02686872/V5*N10/EC.S/1